SoftQuad

H.i.P

HoTMetaL
INTRaNET PUBLISHER

SoftQuad

MANUAL ONE

Using
SoftQuad HoTMetaL intranet Publisher 1.0

Notice

Contents

Introducing SoftQuad H.i.P.

Welcome to SoftQuad HoTMetaL intranet Publisher (H.*i*.P.), a tool for creating, maintaining, and monitoring your company's intranet.

What is SoftQuad H.i.P.?

SoftQuad H.*i*.P. is an integrated package of applications that help you easily build and maintain an intranet site. SoftQuad H.*i*.P. uses advanced HTML (Hypertext Markup Language) to create a more dynamic and powerful way of sharing information. Users can create customizable tables of contents for documents, create their own HTML elements to enrich the document structure, and use cascading style sheets to enhance the style and presentation of the information. SoftQuad H.*i*.P. also uses several of its own extensions to HTML to produce pop-up windows, document annotations, and one-to-many hypertext links.

The SoftQuad H.*i*.P. package consists of four components:

- The *H.i.P. Information Manager*: the command center for SoftQuad H.*i*.P.
- The *H.i.P. Viewer*: plug-ins that let you read documents in H.*i*.P. format using your Netscape Navigator or Microsoft Internet Explorer Web browser.

- The *H.i.P. Editor*: create and edit individual H.i.P. or HTML documents.
- The *H.i.P. Monitor* keeps watch over your intranet for specified events.

These components work closely together to allow you to create, manage, and view shared information over your intranet.

About this tutorial

This tutorial is divided into four main sections:

□ *Introducing SoftQuad H.i.P.* (this section)

□ *How H.i.P. Helps*: a fable about what H.i.P. can do for an average company. This section isn't required reading, but it summarizes the main features of SoftQuad H.i.P. in a real-world scenario.

□ *How Hal got H.i.P.*: the H.i.P. tutorial. Follow in Hal's footsteps as he learns about SoftQuad H.i.P.

□ *H.i.P. Editor tutorial*: a special tutorial focused on the H.i.P. Editor and using HTML.

There is also a separate SoftQuad H.i.P. Reference Guide which details the functionality of each of the components of SoftQuad H.i.P.

Because SoftQuad H.i.P. documents are based on HTML, it is important that you understand at least the fundamentals of HTML in order for you to use SoftQuad H.i.P. The reference manual has a chapter called *Core HTML*, which is a general introduction to the basic features of an HTML document. You will also find a section in this tutorial on HTML on page 135.

In addition, the H.i.P. Editor's on-line help contains a detailed reference guide to all of the HTML elements and attributes. Choose the **HTML Reference Guide** command in the H.i.P. Editor's **Help** menu.

Windows 95 and Windows NT

SoftQuad H.*i*.P. runs almost identically under Windows 95 and Windows NT. The illustrations of dialog boxes and document windows in this tutorial show H.*i*.P. running under Windows 95. The layout and controls are the same in the corresponding dialog boxes under Windows NT. In this manual, the term *folder* is used for what Windows NT calls a *directory*. Any other platform-specific information is noted in the appropriate sections in the manual.

Using the tutorial

Throughout this tutorial, steps that you need to carry out in order to complete a task are marked with a solid , square bullet, i.e.:

■ Do this!

You will occasionally see a marginal icon like the one at left. This highlights a tip or notable feature of H.*i*.P. mentioned in the adjacent text.

Throughout this manual, the terms 'left' and 'right' mouse buttons refer to the default mappings of the mouse ('main' and 'secondary'). If you have a left-handed or non-standard mouse, please translate these terms into whatever vocabulary is appropriate for you.

The tutorial is thoroughly indexed. We have tried to anticipate the various keywords under which users may try to look up a topic, but sometimes you may have to try more than one before you get a 'hit'.

On-line help

This tutorial is available on-line, by choosing the **H.i.P. Help** command from the **Help** menu of any H.*i*.P. component, then choosing the tutorial from the opening page of the on-line help. The on-line help is in H.*i*.P. format; to be able to use the help documents, make sure you have installed the H.*i*.P. Viewer plug-ins for your default Web browser.

When you choose **H.i.P. Help**, the H.*i*.P. Viewer will be launched (if it isn't already), displaying an appropriate help file.

The help files have four Live TOCs (live tables of contents), which are displayed in the Live TOC portion of the H.*i*.P. Viewer window. There are three 'user defined' Live TOCs: **Main Heads** (this is the default, and displays the document headings non-hierarchically), **Commands** (displays all commands referred to in the file), and **Elements and Attributes** (dis-

plays all HTML elements and attributes referred to in the file). In addition to these, the H.*i*.P. Viewer defines a Live TOC called Full Tree that displays the document headings hierarchically.

You can switch views by clicking on the ▦ toolbar button and choosing a Live TOC from the pop-up menu that appears. Commands and Elements and Attributes may be blank for certain documents, depending on their contents.

The help files also have three *views* called Full, Beginner, and Intermediate. Full is the default and displays the entire contents of the file. You can switch views by clicking on the ▦ toolbar button and choosing a view from the pop-up menu that appears. These views may show identical content for some files. Note that clicking a link whose target is contained in a hidden part of a file will not be successful. If this occurs, you should choose the Full view for the file being linked to and try the link again.

See also the file *readme.wri* for last-minute information on SoftQuad H.*i*.P. This file is in the SoftQuad H.*i*.P. folder and is also available from the SoftQuad H.*i*.P. program group.

A whirlwind tour of SoftQuad H.i.P.

The following sections will give you a taste of the SoftQuad H.i.P. user interface, and an idea of the purpose and functionality of each component of the SoftQuad H.i.P. package. Once you've figured out what SoftQuad H.i.P. is and does, you can move on to the tutorial to see it in action.

The H.i.P. Information Manager

The H.i.P. Information Manager is the primary environment for working in SoftQuad H.i.P. From the Information Manager you can create, import, and organize intranet *projects* (groups of pages—usually organized by topic or user). You can manage and view projects in different ways:

- [ ] Add, edit, delete, and move H.i.P. document files.
- [ ] View projects by files and folders, or with a map of links.
- [ ] Find broken links and orphaned files.
- [ ] Publish entire projects to your intranet.
- [ ] Find, or find and replace text throughout entire projects.
- [ ] Map external World Wide Web sites.

The Information Manager is also the launch pad for all of the other components of SoftQuad H.*i*.P. From here, you can launch applications to create:

☐ *Live tables of contents* (Live TOCs) for your intranet documents

☐ Cascading *style sheets* and custom document *views*

☐ *User-defined extensions* (UDEs)—custom HTML elements that *you* create

You can also launch the H.*i*.P. Editor and the H.*i*.P. Viewer from the Information Manager.

The Information Manager interface

To get started with creating, modifying, or managing a H.*i*.P. project, launch the Information Manager from the SoftQuad H.*i*.P. program group. Since you haven't loaded a project, the Information Manager window will be empty. The Information Manager interface consists of *panels*—divided sub-window areas, a *toolbar*, and *menus*.
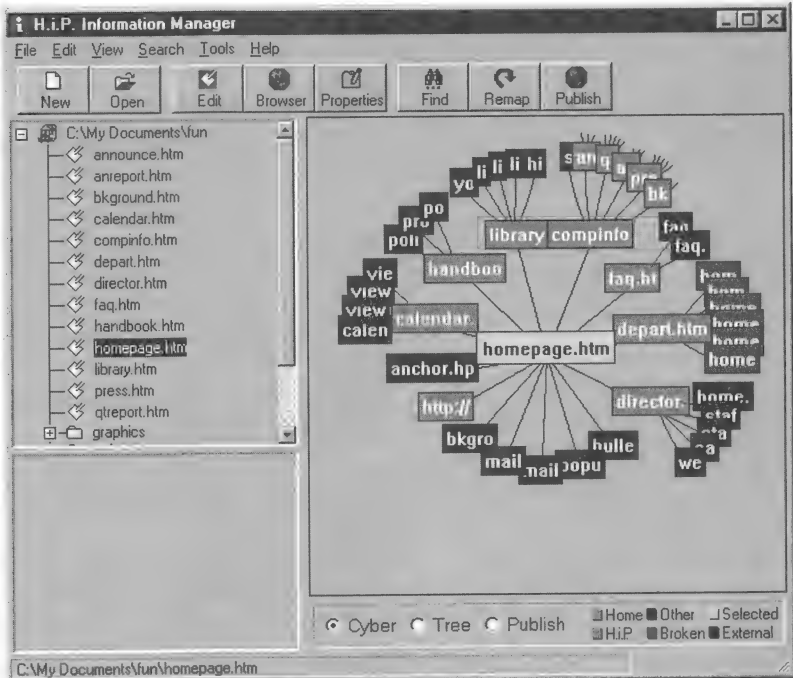
Information Manager with a simple project

Panels

The H.i.P. Information Manager window has three panels which show the same H.i.P. project in different ways:

1. The *Link* panel, showing hyperlinks between files, occupies the right side of the H.i.P. Information Manager window. The Link panel can show one of three different *displays*: the *cyberbolic* display, the *tree* display, and the *publish* display.

2. The *Project* panel, showing the files and folders within the H.i.P. project folder, occupies the upper left portion of the H.i.P. Information Manager window.

3. The *H.i.P. Pocket* panel, which can display user-selected groups of files, is in the lower left.

When you click on a file in any panel, the corresponding file will be selected in the other panels. The full sequence of folders containing the selected file will be shown in the status area at the bottom of the Information Manager window.

Toolbar

The toolbar buttons provide access to the most important commands. A *tooltip*—a short description of what a toolbar button does—appears above the toolbar buttons if you move your mouse pointer over them.

You can show or hide the toolbar by choosing **Toolbar** from the **View** menu. You can switch between small icon toolbar buttons and large toolbar buttons with labels by choosing **Toolbar Labels** from the **View** menu. A checkmark appears beside **Toolbar Labels** if toolbar labels are selected.

☐ *New*: launches the template wizard, guiding you through the creation of a new H.i.P. project.

☐ *Open*: opens an existing H.i.P. project.

☐ *Edit*: launches the appropriate editor for a selected file; i.e., the Live TOC editor if the selected file is a TOC definition list; the H.i.P. Editor if the file is an *.html* or *.htm* file, etc.

☐ *Browser*: launches the H.i.P. Viewer (your Web browser—Netscape Navigator or Microsoft Internet Explorer—for which the H.i.P. Viewer plug-ins have been installed) and opens the selected file.

☐ *Properties*: launches the Document Properties dialog box (using the selected file); in this dialog you can create and modify many different aspects of H.i.P. documents, including user-defined extensions, live tables of contents, and cascading style sheets. .

☐ *Find*: lets you search and replace text across a project.

☐ *Remap*: rebuilds the different project displays and shows any changes that have been made to your project.

☐ *Publish*: lets you move your documents to the Intranet server and reorganize your links.

Menus

All commands can be chosen from the menus. The following list gives a brief overview of the functionality in each menu of the H.*i.*P. Information Manager.

☐ **File** menu: Creating, opening, and importing H.*i.*P. projects; creating and importing files; launching the H.*i.*P. Viewer and appropriate editors for H.*i.*P. files; exiting.

☐ **Edit** menu: Copying, renaming, and deleting files.

☐ **View** menu: Moving between different link displays; controlling toolbar and panel visibility; setting preferences (options).

☐ **Search** menu: Finding and replacing text; finding broken links.

☐ **Tools** menu: Editing Live TOCs, User-defined Extensions, and styles; Publishing; configuring a proxy server; mapping a remote site; rebuilding the link map.

☐ **Help** menu: View on-line help.

For a detailed reference on the H.*i.*P. Information Manager please see the *SoftQuad H.i.P. Reference Guide*. This tutorial also contains detailed instructions on using the H.*i.*P. Information Manager.

The H.i.P. Viewer

The H.*i.*P. Viewer consists of plug-in files for your Netscape Navigator or Microsoft Internet Explorer Web browser. The plug-ins create dynamic *live tables of contents* for any H.*i.*P. document that you load into the browser, and allow users to view H.*i.*P. *pop-ups* and *one-to-many links* (multilocs).

> **Note** The H.i.P. Viewer uses JavaScript to display H.i.P. documents. If it isn't already enabled, you must enable JavaScript in your browser in order to use the H.i.P. Viewer. See your browser documentation for information on enabling JavaScript.

Loading a H.i.P. page

Because the Viewer component of SoftQuad H.*i*.P. works with your existing browser, the H.*i*.P. Viewer plug-ins will automatically be used when you load a H.*i*.P. file. You can also launch the Viewer from the H.*i*.P. Information Manager by clicking on the document you want to view in the Project panel or Link panel, and doing one of the following:

- Choosing **View in Browser** in the **File** menu.

- Clicking on the (launch browser) toolbar button.

- Right-clicking and choosing **View in Browser**.

This will launch your browser with the H.*i*.P. Viewer plug-in and load the page that you selected. From the H.*i*.P. Editor, you can display the current H.*i*.P. document in the Viewer by using the **Preview** command.

When you load a H.*i*.P. document in the Viewer, the window splits into two panels. The left panel is the table of contents panel, and the right is the document panel. The table of contents panel contains a *Live* (i.e., dynamic) *Table of Contents* (Live TOC) generated from the document. The table of contents panel has a toolbar at the top. The document panel displays your H.*i*.P. document the way your browser normally displays HTML documents, but supports some of the special H.*i*.P. extensions (pop-ups and multilocs) as well.

Live tables of contents

A Live TOC generally consists of the text of the headings (H1-H6 elements) in the document. If you click on an item in the Live TOC, the view in the document panel will jump to the corresponding location in the document. You can adjust the Live TOC's levels of nested headings by clicking on the ⊟ (contract) or ⊞ (expand) icon next to the Live TOC entry. The TOC is fully expanded by default.

The toolbar

Using the H.i.P. Viewer involves more than just browsing HTML documents. From the toolbar at the top of the table of contents panel you can:

- [ ] **Change Live TOC:** choose a different table of contents, create a new table of contents, or edit an existing table of contents (based on different elements in the document) using the Live TOC editor.

- [ ] **Sort the Live TOC:** sort the live table of contents in either ascending or descending alphabetical order. This re-orders only the Live TOC panel and not the document panel.

- [ ] **Expansion Level:** select the level of nested headings and subheadings displayed in the table of contents.

- [ ] **Change View:** choose author-defined alternate views of the document. For example, a technical manual may have views for introductory, intermediate, or advanced information.

- [ ] **Search:** perform a context-sensitive search on the document. You can search for words or patterns within elements.

- [ ] **Annotate:** create and read notes or comments about the document. You can view annotations made by other people, or make your own.

- [ ] **Subscribe:** instruct the H.i.P. Monitor to notify you when a specific page, pages by a particular author, or pages on a particular topic have been updated.

- [ ] **Options:** select options for the H.i.P. Viewer, (e.g., default Live TOC expansion level, user name, e-mail address).

- [ ] **Help:** view the on-line help (in H.i.P. format).

The document panel

In the document panel you can view special H.*i*.P. features such as pop-up windows and one-to-many links (multilocs).

☐ Pop-ups can appear either inline (as part of the text) or block (in a separate window).

– Inline pop-ups appear as 🗩 icons in the document panel. Clicking on an inline pop-up opens a plain text pop-up window that will disappear if you click anywhere else on your screen.

– Block pop-ups appear as block icons (on a separate line from other text). The block pop-up icon can be any image the author chooses, but the default is:

Clicking on a block pop-up opens a new browser window that can contain HTML, images, links, etc. This window will remain open until you close it by double-clicking on the control button in the top left corner.

☐ A one-to-many link (multiloc) is a menu of links. A multiloc appears as a small ➡ icon in the H.*i*.P. Viewer. When you click on the icon, a pop-up menu will appear, displaying all of the links accessible from that single location. Choose the link you want to go to. Some items can *fly-out* to reveal sub-items.
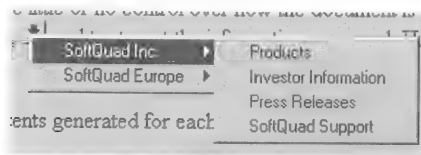
The H.i.P. Editor

While the H.i.P. Information Manager gives you a project-wide view, the H.i.P. Editor is for creating and editing individual HTML or H.i.P. files. Based on SoftQuad HoTMetaL, the H.i.P. Editor is a professional HTML authoring tool with additional capabilities for creating documents for SoftQuad H.i.P.

The H.i.P. Editor integrates all the power of a word-processor with the tools for creating proper HTML documents for the Web or an intranet. Using the H.i.P. Editor, you can:

☐ Create new H.i.P. documents, edit existing H.i.P. or HTML documents, or import documents from popular word processing formats and convert them to valid HTML.

☐ Use H.i.P. features such as user-defined extensions, pop-ups, and one-to-many links.

☐ Validate your documents for both HTML conformance and for accessibility to users with disabilities.

☐ Create and edit graphics for your documents using the MetalWorks graphics editor and image mapper.

☐ Launch the **H.i.P. Document Properties** dialog and the H.i.P. file editors (e.g., Live TOC editor).

Launching the H.i.P. Editor

Launch the Editor from the Windows Start menu or by double-clicking on its icon in the SoftQuad H.i.P. program group. You can also launch it from the H.i.P. Information Manager by selecting the *.htm* or *.html* file or template you want to edit, then clicking on the (**Edit File...**) toolbar button, or by right-clicking on the filename in the Project panel and choosing **Edit File...** from the pop-up menu, or **Edit File...** from the **File** menu. This launches the H.i.P. Editor and loads the file or template that you chose.

Overview of menus

This section provides a summary of the main features.

☐ **File** menu: file manipulation; e.g., opening and saving files, converting to and from H.i.P. format, setting document properties.

☐ **Edit** menu: cutting and pasting, find and replace, spell checking, merge/remove annotations.

☐ **View** menu: local display formatting; show and hide tag items; view source file.

☐ **Markup** menu: inserting and changing markup, creating and editing anchors.

☐ **Format** menu: text style, alignment, and color; backgrounds.

☐ **Tools** menu: image, table, and frame editing; accessibility tools; find and replace URLs.

☐ **Forms** menu: inserting and editing forms.

☐ **Special** menu: checking document conformance, creating and running macros, options.

☐ **Window** menu: appearance and control of document windows.

☐ **Help** menu: on-line help, other help documents.

For more information

To help familiarize yourself with the H.i.P. Editor, we suggest that you try the H.i.P. Editor tutorial (see page 135), which has exercises demonstrating the main features of the H.i.P. Editor, and HTML.

The H.i.P. Monitor

The H.i.P. Monitor is a system administration tool that keeps watch and tests your intranet server, operating system, server clients and their pages 24-hours a day. The H.i.P. Monitor works with the other components of SoftQuad H.i.P. to allow you to publish and remove pages at a specified time, and to inform users who subscribed to certain pages through the H.i.P. Viewer that those pages have changed.

The H.i.P. Monitor can detect and report such things as:

☐ *Server software events*: server down, network process down, suspicious requests.

☐ *Server operating system events*: disk too full, root login, user login.

☐ *Client events*: page expired, broken links, page effective, replace page, project published.

Launching the H.i.P. Monitor

The H.i.P. Monitor is accessed through a browser. When you launch the H.i.P. Monitor Event Browser or the H.i.P. Monitor configuration tool from the SoftQuad H.i.P. program group, you are actually loading an HTML document into your browser. All of your work in the H.i.P. Monitor is done by filling out HTML forms that are processed by a program on your intranet server.

Of course, the administration pages are not 'public'. They're your pages and require a password to access them. Once you have installed the H.i.P. Monitor on your intranet server, you can surf to the Monitor homepage:

■ Start the H.i.P. Monitor from the H.i.P. Program Group.

When your browser appears, it will present the H.i.P. Monitor's homepage.

■ Place the homepage in your **Bookmarks** or **Favorites** list.

Much or what the H.i.P. Monitor does requires reporting to you, the administrator. You can read the Monitor's weekly reports, as you might expect, by surfing to the Monitor's intranet page and clicking on the appropriate week. Here you will find a list of everything (within the Monitor's responsibilities) that has happened during the week: operating system alerts, suspicious requests, root logins, broken H.i.P. page links— you name it. You can look at a weekly report now, though, if you have just installed the software, there shouldn't yet be any events recorded.

■ Click on **Browse Events**.

The **Browse Events** page presents a table of dated events that have occurred in the current Monitor week (beginning Sunday night). You can move from week to week by clicking on PREVIOUS WEEK or FOLLOWING WEEK. Clicking on these now will have no effect because there isn't yet a history of events to view. Return to the H.i.P. Monitor homepage (actually its menu page) by clicking on the word MENU in the banner that spans the top of your page.

A note about file formats

Although H.i.P. documents are HTML documents, there is some special HTML markup included in a H.i.P. document so that the H.i.P. Viewer will display the document with all the H.i.P. Viewer features. This markup consists of:

– a SCRIPT element in the documents' HEAD element, containing a *JavaScript* program

– a FRAMESET that defines the frames displayed in the H.i.P. Viewer

– a NOFRAMES element that contains the BODY of the original document

– a link to an auxiliary file called *epsilon.hpv* in the same folder

This special SoftQuad H.i.P. markup is hidden from view in the H.i.P. Editor. While you can open the file with a text editor to see what the markup looks like, *do not edit* the H.i.P. sections of the document—it will not load properly in the H.i.P. Viewer. You can learn the technical details (useful if you are automating intranet publishing) about the file formats used by SoftQuad H.i.P. in the *File and markup formats* appendix to the SoftQuad H.i.P. Reference Guide.

You can use plain HTML documents in your H.i.P. projects, but in order to view the document with H.i.P. features—such as Live Tables of Contents, views, and annotations—in the H.i.P. Viewer, you need to convert documents to SoftQuad H.i.P. format. This does not alter the content of the document; 'H.i.P.-ification' just adds some special HTML markup that tells the H.i.P. Viewer plug-ins to start when you load the document into your Web browser. You can H.i.P.-ify your documents on a project level

from the H.*i*.P. Information Manager or as you create and edit them in the H.*i*.P. Editor.

How H.i.P. helps

A story that happened in the recent future...

EG Construction Inc. builds office towers, hospitals and factories throughout Canada, the United States and Britain. Their business method is to design, build and, in some cases, operate a building on the client's behalf. More than 700 people work in EG's head office and a similar number, many of them contract and temporary employees, work in construction field offices and facilities management offices.

Elizabeth works at head office in the mechanical engineering division.

Sharing information...

As is the case with any engineering firm, cost control at EG depends heavily on the coordination of plans and specification documents among several departments. While, in theory, each department was to have a person assigned to provide information to other departments, in practice, none wanted to expend the person-hours. Everyone's time was billed, hour-by-hour, to each contract, and any intra-company cooperative effort showed up immediately as an increase in contract-allocated cost. Liz and people in her position in other departments were caught in the middle.

They responded by suggesting a new way to share engineering information: specs and drawings would be posted on the internal network for anyone to see. Of course, their first attempt was rejected. For one thing, the specs were written in three different formats so no one could depend on making any sense of a particular word processed file.

However, the marketing group encouraged the basic notion. They were charged with creating complex proposals for competitive bids. While these were based on technical documents, the proposals themselves were carefully designed and typeset, and heavily illustrated with drawings and site photographs. They pointed out that, at the very least, they would like to know when new material was available, and where to find relevant old material to support similar project bids.

Liz and her cohorts went back to their drawing boards. Rather than delivering all the specs and drawings for everyone to view, they decided to try another approach. They proposed a system that offered the services of a library card catalog, a news clipping bureau, a press-release agency, a department store mail order catalog, and a post office.

The system is still evolving, but it's easy enough to describe; a typical Tuesday morning is a good place to start.

Monitoring information...

Liz tries to get in a little early on Tuesdays because it's weekly timesheet day at EG Inc. It's not that she's any more concerned with proper cost allocation than her co-workers, but filing the sheets has changed. Timesheets are now filled out on a Web page, viewed in a browser. The browser has been upgraded to work as the H.i.P. Viewer, so it uses pop-ups full of explanations for each field in the timesheet form.

But that's not what gets Liz in early. The SoftQuad H.i.P. monitoring functions have been set up to e-mail Brian in the accounting department when timesheets are filled out. He runs a lottery for everyone who files on time and Liz has won a dinner out for herself and her husband twice this year. As soon as Brian makes the draw and posts the winner's name on a H.i.P. page, email goes to every subscriber in the company.

Navigating a document...

After submitting her timesheet, Liz moves on to the structural engineering division's page. This is a good example of how her original idea has made her job much easier. The page itself is very simple: just a list of engineering project drawing and document component names. The file is straight Hypertext Markup Language (HTML, the *lingua franca* of the Web). But it's a very long list. Liz has looked at this list in a non-H.i.P. browser and it's like reading several pages of a phone book—except that it's not sorted.

In her H.i.P. Viewer, the list has some useful structure. The list itself is displayed in the large, right-hand section of the browser window. On the left, in a smaller section, is a Live TOC—a table of contents that is produced on-the-fly. There are only a half dozen of these projects that Liz is concerned about, so she chooses an alphabetical sort for the Live TOC—which now looks more like an index. She finds the first project she's looking for and clicks on it in the index. It appears on the right.

Some new markup...

The line of text that labels the drawing Liz is interested in has a new pop-up. She clicks on it. Another browser window opens and there is a brief explanation of changes that structural will be making to the depth of the roof webs over an area of a new plant. There is a link, right in the middle of the window that, when Liz clicks on it, becomes a menu of several possible destinations.

Liz e-mails the manager of the team that made the alterations—his e-mail address is on another page in their web site, but she put the page in a H.i.P. list when she first saw it. She clicks on a line in the lower left section of her browser and a list, containing the pages that have links to e-mail addresses, pops into view. She clicks on the team manager's name and an e-mail editor appears where she types her message.

Organizing a
H.i.P. project...

Next, she opens the H.i.P. Information Manager. She's the site editor for the mechanical department's pages, so she finds the page with the information about the conveyor installation. With SoftQuad H.i.P. this is easy. The cyberbolic display shows her an image of her H.i.P. project that always reminds her of fireworks: a cluster of long, thin trails bursting out of interconnected nodes.

With a right mouse click, a menu offers the H.i.P. Editor, which opens with the page in view. The rest of the members of the mechanical department will have to refer to this page, of course, but Liz is more concerned that the electrical people see it too. They will be pulling their parts and supplies inventory together, expecting to be at the site next week.

Some more
new markup...

In the schedule section of her procedures document, Liz places a new HTML element—actually, it's a user-defined extension that was designed by the systems administrator when the intranet was installed and SoftQuad H.i.P. was chosen as the HTML platform. Liz sees it on her H.i.P. editor screen as a tag labeled 'Schedule Warning'.

Just to make the problem clear, she adds a multi-location link (a 'one-to-many' link) that will appear as a nested menu of destinations when her page is browsed.The electrical people will be able to see the problem for themselves by viewing the same elevation drawing fragment that alerted her; they will have a nested menu of link choices to the original files at the structural divisions site.

Different ways of looking at documents...

Since she isn't sure yet whether department managers have made any decisions, she places a 'Manager's view' around her description of the problem. Very likely, she will change this view so that everyone can see it, but, until the managers have made up their minds, other engineers don't have to be involved.

Keeping everyone informed...

It's everyone's separate responsibility to subscribe to the appropriate pages and use the information to keep their engineering projects in line. But, just to be sure, Liz adds a distribution list to her schedule page. Once she has saved it, the H.i.P. Monitor will send out e-mail to everyone she has named, notifying them of a change to an important page.

Consistency and completeness...

Before she closes the conveyor installation project, she notices that the contact information page would be a good beginning point for a similar page she has to create in another H.i.P. project. In the H.i.P. Information Manager, she selects the page and drags a copy to a new location within the 'Machinery Installations' directory tree. The H.i.P. Information Manager automatically notes some unconnected links on the new page and Liz is able to reconnect them and check that nothing has been left dangling. However, some links in this second project appear to have broken: they show up as blue document titles in the cyberbolic display. Some of the pages elsewhere in the EG's intranet have been moved or deleted by other divisions. Liz sends a couple of e-mail messages, one to a construction office in Britain, asking whether the information has been moved or canceled.

This second installation project has the same structure, and many of the same links, as the first one. In fact it was created with a template for such mechanical installation projects so it went together very quickly. Liz did the template herself, but templates in other divisions are often set up by the systems people rather than the local division editors.

Converting document formats...

Later in the morning, Liz receives a document that has been sent on a disk by courier. It's information regarding a new pricing schedule from a parts supplier. Rather than print such a long document and hand it around the office, Liz converts it from WordPerfect 5.1 format—a 'standard' at one time in the civil engineering business—into H.i.P. format. As a H.i.P. HTML file, the price list is much easier to use and a lot less expensive to distribute.

Searching for information...

The Live TOCs will index the parts once Liz has marked them. Since pricing information is distinct from dimensional information, she usually creates two views of these documents—one for the designers and one for the estimators. Anyone can flip back and forth between these two views and the designers have begun to pay more attention to the price list.

The biggest advantage, however, is that everyone in mechanical (and other divisions too, if they are interested) can search a particular part name and have the results of the search highlighted on the screen. The estimators use this technique to do their comparison shopping; they put competing price lists in separate browsers and run their comparisons to find the best price.

While the conversion is easy, Liz would rather receive the price lists (and other supplier documents as well) in H.i.P. HTML, and she is trying to persuade EG to make this known to suppliers. In the meantime, a few suppliers have posted their information on their own Web servers in ordinary HTML and these are functional, if not as convenient when viewed with the SoftQuad H.i.P.-equipped Netscape and Internet Explorer browsers.

Since H.i.P. documents can contain keywords, Liz has made sure that all of the projects she creates are well-described by their H.i.P. topics. This allows her to do easy searches on her own division's projects. SoftQuad H.i.P. prompts for this information whenever she edits a new file.

Staying up to date...

Toward the end of the day Liz creates a new H.i.P. document for the EG employee newsletter. The newsletter is no longer printed on paper, of course. It can contain a lot more information and it has no rigid deadlines. Her division hosts a lunch-hour design seminar for the rest of the company and Liz wants to create an event notice and sign-up form. Since the sign-up has a deadline, she sets an expiration date on the form page, and creates another page that will inform late-comers that they have missed their chance. The second page will automatically replace the sign-up form when the date arrives. The information page will simply expire on the day after the event.

One of the engineers in the office was married on the weekend so Liz adds this news on a page for the social section of the newsletter. These pages are often added to EG's public newsletter for clients and suppliers. Viewed with H.i.P. browsers, the pictures will be positioned within a pop-up. Those who want to can skip them.

Some years later...

Thanks to Elizabeth's solution to the file-finding problem so many years ago, EG was able to cut their costs and increase their margins as their SoftQuad H.i.P. use evolved and grew into an office operating environment. They soon grew large enough to bid for the construction of the Bering Strait Ice Bridge: and they won. The 'greatest engineering challenge of the 21st century' brought to an end the long separation of Asia and North America and spurred the creation of the Free Republic of Aleutia, the world's fastest growing post-information economy. Today, the original specification and drawings of the project are available at the Bering Strait Ice Bridge Archive Initiative. The H.i.P. Viewer, version 12.0, is recommended.

How Hal got H.i.P.

SoftQuad H.i.P. provides a new way to edit, organize and distribute information across your company's intranet. If you are new to Web pages, SoftQuad H.i.P. will make your introduction much easier. If you have already browsed Web pages, many aspects of SoftQuad H.i.P. will be familiar. For example, H.i.P. documents are viewed with your Netscape Navigator or Internet Explorer browser.

You may have written some HTML (Hypertext Markup Language) pages already. If that's the case, you will find the H.i.P. Editor easy to learn. You may be familiar with it—it's based on the same technology as HoTMetaL PRO. But it has some entirely new features that will make your documents easier to find, easier to read, and easier to understand.

This tutorial, 'How Hal got H.i.P.', is the not-quite-true story of a new employee who was charged with the responsibility for what seemed, at first, like a daunting information project—keeping the employees of a manufacturing company informed of a complex corporate merger and move. Establishing a corporate intranet was the obvious choice. With good navigational assistance, employees could find their way in even the largest repository of news, policy, site maps, schedules and instructions. The information could change day by day, receiving contributions from every department. With the H.i.P. Information Manager, a site publisher, our man Hal, could keep it all on track.

Hal had a lot to learn about his new company, Exemplar Manufacturing Inc. and its new acquisition, Priority Machining. As a new employee, he knew how difficult it could be for any employee, new or long-term, to learn the basic business of filling out company forms, following reporting

procedures and generally just getting around. Of course, he also had to figure out SoftQuad H.i.P. But that was the easy part. Before long Hal knew how to:

☐ Create a new project with the H.i.P. Information Manager (see page 39).

☐ Write and edit new H.i.P. pages with several novel features:

- pop-ups (see page 101)
- different views for different people (see page 88)
- cascading style sheets (see page 111)
- multiple links to other pages (see page 61)
- specialized tables of contents that can change on-the-fly (see page 83).

He also learned to:

☐ Get a head start with H.i.P.'s extensive template collection (see page 114).

☐ Create an automatic document updating system (see page 132).

☐ Add his own new tags to the HTML tag set (see page 75).

☐ Alert specific groups of readers to new information (see page 122).

☐ Convert documents to H.i.P. format (see page 56).

☐ Keep track of the whole project (see page 30).

Like most of us, Hal didn't have much extra time for getting up to speed—he preferred to figure out just what he needed and keep the job moving. So this tutorial doesn't exhaust every topic. If you're ready to explore more deeply than Hal you should consult the H.i.P. *Reference Guide*.

Here's Hal's situation

Exemplar Manufacturing Ltd. grew and acquired another company, Priority Machining. The new, merged company would soon be moving to a brand new plant and office complex in the town of Neoteric, New York. As Hal began to work on his SoftQuad H.i.P. intranet site, employees were located in three different towns, reported to 35 different managers, and had all of the anxieties you would imagine when 400 employees, with their production machinery, offices, computers, and favorite coffee mugs had to move to a new town and learn to work together. Let's see whether Hal and H.i.P. were able to help.

On a morning that he won't soon forget, Hal admitted to his boss that he knew a little about Web pages. "Excellent," his boss remarked, "you can get all of this information out on our corporate intranet. It'll be up-to-date, reassuring, informative, extensive, inexpensive and colorful and ... and ... "

"Accurate?" offered Hal.

"That too. Yes. Get your ducks in the pond and get back to me."

Naturally, Hal had several worries. Did Exemplar actually have a corporate intranet? Where was this information going to be stored? How would he keep track of it, get it approved by the 12 members of the Exemplar's moving management committee, and keep it accurate? Were there any templates that he could use to get started? How would he present the same information to people with completely different moving problems? And did Neoteric have a park with a duck pond? (Hal was partial to ducks.)

Fortunately, Exemplar Manufacturing's systems people had installed an intranet server and were upgrading everybody's browser with the H.i.P. Viewer plug-ins. They had also installed the H.i.P. Information Manager and Editor on Hal's PC, and the H.i.P. Monitor was running on the server. The H.i.P. Viewers were also being installed at Priority where some of the systems people were already familiar with HTML. Hal took a deep breath.

A first look at the H.i.P. Information Manager

Hal had already viewed a lot of Web pages from his home computer, so he knew about clicking on links and surfing to other pages; the usual browser stuff was nothing new. When he began the project he had never created an HTML file, let alone a H.i.P. HTML file, but he assumed they had to be written and edited somehow—so the H.i.P. Editor made sense. But he didn't really see the necessity of the H.i.P. Information Manager. In fact, it wasn't until several days later when Hal had created about a dozen interlinked files, that he began to see how the H.i.P. Information Manager could help. By the time the whole project was in place—all 220 pages with their graphics, hundreds of links, H.i.P. Views, and new user-defined extensions... well, by that time Hal practically spent his day in the H.i.P. Information Manager.

In the meantime, Hal decided he would start by just having a look at one of these H.i.P. documents. He had a browser on his desktop and he was just about to give a click when he remembered what Marco, the systems guy, had said: "Hal, just do this for me." But what was it Marco had said after that?... Something about using the H.i.P. Information Manager as a starting point. He clicked on the H.i.P. Information Manager icon with a certain suspicious reluctance. Try it yourself.

What Marco had said was that the H.i.P. Information Manager is like an air traffic control tower: you may not have actually built the pages, but, from this vantage point, you can keep the connections working and prevent things from bumping into each other. Let's get a few blips on the screen.
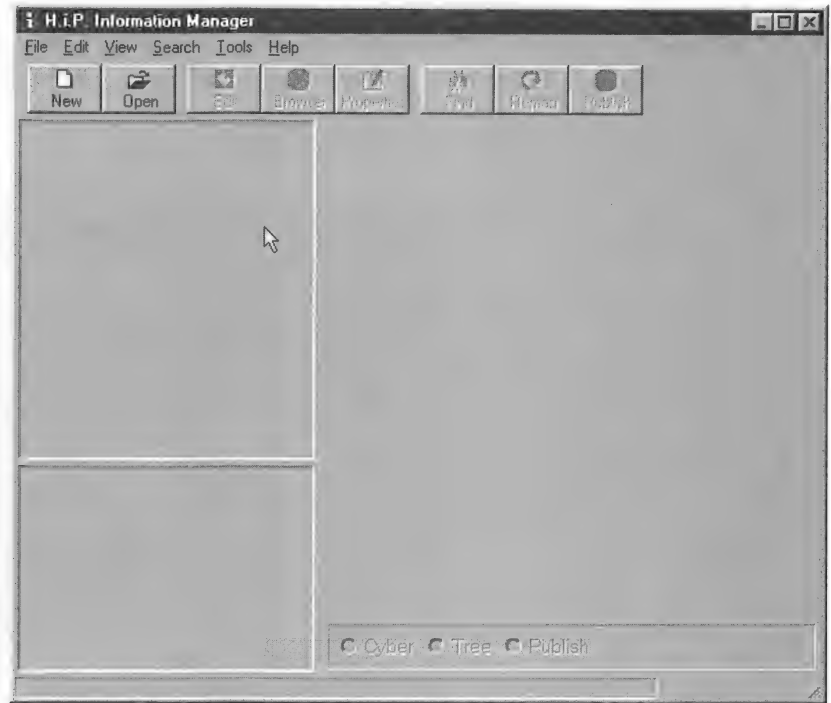
What Hal saw was a new application window with H.i.P. Information Manager in the title bar. There were two gray panels on the left and a large, gray area on the right. Only two of the buttons were active: New and Open. But there were six others across the top and three radio buttons at the bottom.

To see what a project looks like when it has acquired several dozen files, Hal imported a sample project called the Underground, and had a short tour with the cyberbolic display as it appeared on the right side of the H.i.P. Information Manager. The panel where the cyberbolic display was displayed is called the Link panel because it's mainly for viewing the link relationships among large numbers of files.

Of course, by now, Hal is an accomplished intranet publisher.

To follow in Hal's footsteps:

- Choose Import Project... in the File menu.

A dialog called **Select Project Home Page** appears.

- Navigate to the *help* folder where you will find a *tutorial* folder. In the *tutorial* folder there is a folder called *undrgrnd*. Select it.
- Click on the file *index.htm*.
- Click on the [Open] button.
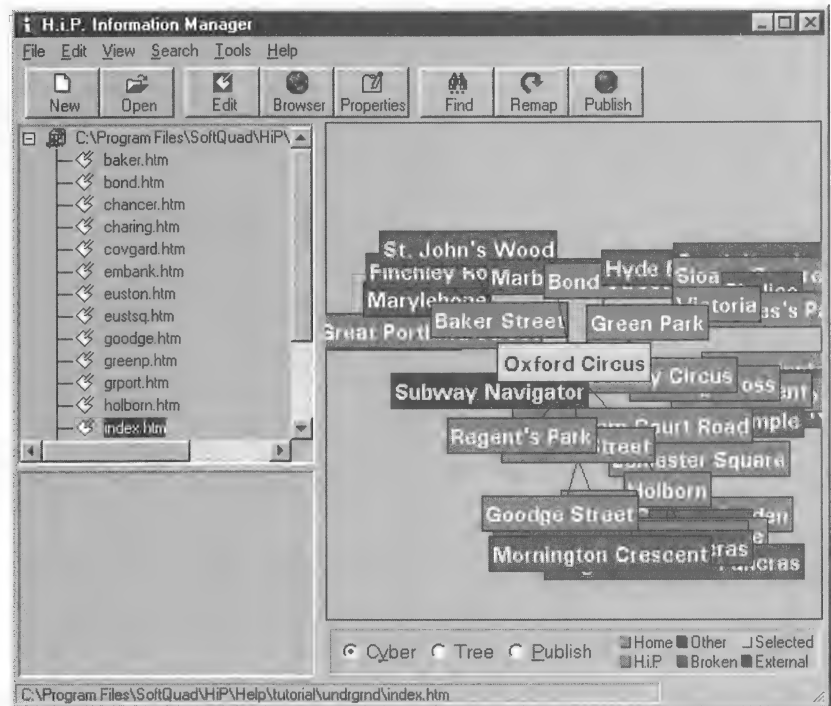
The H.i.P. Information Manager's 'cyberbolic' link display appears in the large, right panel. Hal guessed that he was looking at a representation of part of the London Underground system.

The image reminded Hal of a spider plant. Hal also discovered that, by clicking on the screen with his right mouse button (anywhere but on a document name), he got a menu in which he could make several choices about his cyberbolic display. The choices that you will see when you first look at the menu may vary from the picture below (they are retained from one session to another), so you may want to click on each line so

that you are aware of both the checked and unchecked choices for **Right Align** and **Long Names**, and the difference between **Show Titles** rather than **Show Filenames**, and **Minimal View** rather than **Full View**. Each of these will be discussed.
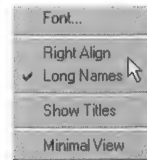
By clicking on **Right Align** (so the check mark appeared), Hal turned his cyberbolic display from a spider plant into peacock feathers, (Hal was partial to peacocks, too). The new image radiated, branched out and radiated again.

Notice how, when you drag one of the 'feathers' with your mouse pointer, the whole network moves to reveal more detail.

- Place your pointer on the document title 'Oxford Circus' and drag it toward the left of the screen.

You can drag successive document titles to find your way right out to the 'Mornington Crescent' at the very tail end of the feather:

☐ Oxford Circus

☐ Warren Street

☐ Euston

☐ Mornington Crescent

(Anyone who has used the London Underground—and many who have not—will know that this is just a tiny fraction of the whole system. Mornington Crescent is no where near the tail end of the real system.)

The names you are reading in the cyberbolic display are probably document titles—unless you've jumped ahead by clicking on **Show Filenames**.

- Try that now by right-clicking on any blank spot on the Link panel and choosing **Show Filenames** from the menu. (If you see the choice **Show Titles**, then the cyberbolic display is already showing filenames.)

With filenames displayed, it's easier to relate names in the cyberbolic display contained in the Link panel to the names shown in the Project display in the upper left part of the H.i.P. Information Manager window.

■ Click on any filename and see what happens in the Project panel.

The same filename is highlighted in both panels. In the cyberbolic display, the name is shown in a yellow box. If you click with your right mouse button on a title or filename in the Link display, you will see a different menu:

■ Try that now by right-clicking on 'Goodge Street' (or, if you are displaying file names, *goodge.htm*) in the cyberbolic display. Don't make a menu choice yet because we're not finished here. But you can see that this is another way to launch the H.i.P. Editor or the H.i.P. Viewer. Click anywhere else to make the menu disappear.

Edit File
View in Browser
Show All Links
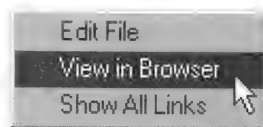
By this time Hal had figured out that the Link display shows links—but there was more to it than he thought. Actually, the cyberbolic display shows 'parent-child' links , but does not show links between 'children'. The linking lines radiate from a file that contains a link (the source file) to the files that are linked to it (the target files). Although the cyberbolic display does not show them, there might also be links from one of these targets to another target—that is, from one child to another child. You should not be surprised later on when, having created a child-to-child link, you don't see any indication of it in the cyberbolic display.

However, the Link display tells you a great deal about the links that it shows: if a link is unusable because the file that the link points at isn't there, the document name (or filename) that should be there, but isn't, is shown in red. There are several such broken links in the London Underground example: 'South Kensington' and 'Hyde Park Corner' are two. A blue box shows a document that is off-site—the *nsubsays.htm* is such an off-site file (its title, in case you are looking at titles in the Link display, is Subway Navigator—if you are curious, right click on it and launch your H.i.P. Viewer). Here's a summary of what each cyberbolic color means:

☐ Grey: a H.*i*.P. file.

☐ Black: a file that is not in H.*i*.P. format.

☐ Green: root file for the cyberbolic display.

☐ Yellow: currently selected file.

☐ Red: broken link. (What appears in the label is a document title or file name of a file that should be there, but is not.)

☐ Blue: external (e.g., World Wide Web) link, outside the local file system.

A different perspective

If you want to see a different perspective on the relations among files

■ Click on the **Tree** radio button at the bottom of the panel (or choose **Tree Link View** from the **View** menu).

This is another kind of Link display of your site and it follows some of the same conventions as the Project display (and Microsoft Windows Explorer) except that it still shows links rather than a folder-and-file structure. A plus ⊞ or a minus ⊟ sign beside a document allows you to expand or collapse that branch of the tree.
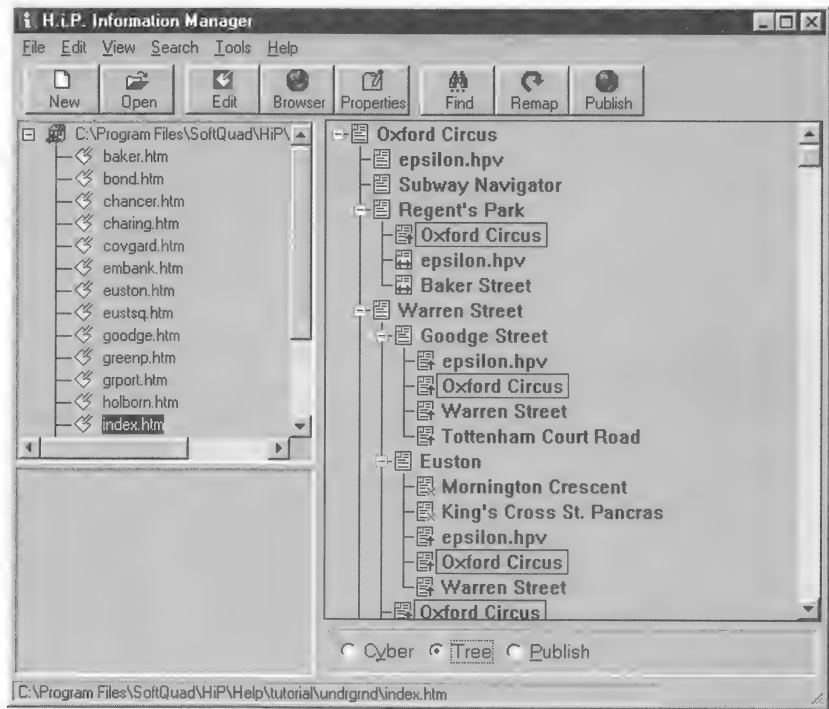
The different icons, shown above, indicate the kind of files that are contained in the project:

☐ Links to pictures (GIF or JPEG images).

☐ Links within the page.

☐ Links to pages that are farther from the root.

☐ Links back to pages that are closer to the root.

As you create auxiliary files that contain other SoftQuad H.i.P. features, you will find that links to these are shown in the Link display. Even though these are not 'viewable' files, they can be edited and that's why they are shown.

Here are the file types you will find in both the tree and the cyberbolic displays:

File extension	File type
.htm	Document
.css	Cascading style sheet
.hpl	Live TOC
.hpa	Annotations
.hpe	User-defined extensions
.hpo	Topics
.hpp	Project

The Project display (the tree-like display on the left side of the H.i.P. Information Manager window), is similar to the display you see in Windows Explorer in that it displays the file and folder structure of the current project. It also indicates (with icons) the condition of links and the existence of orphaned files.
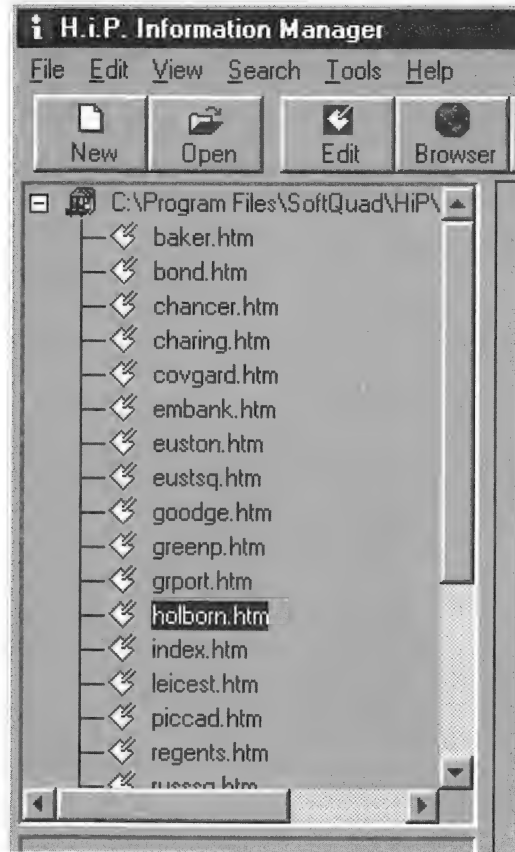
Hal found that, in the Project panel, he could drag and drop files from one place to another in the directory structure. He could also double-click on them to open them in the H.i.P. Editor. However, the icons that appear are special to H.i.P. For an exact definition of their meanings, see the *Interface* chapter of the *Reference Guide*.

He also found that, by clicking on file names in the Project panel (on the left side) the corresponding document title was highlighted in the tree or cyberbolic display. Try it.

Finally, there's a smaller panel on the bottom left. Hal was completely mystified about that one. You too? Well, every story needs a little mystery.

Creating a new project

After his tour of the London Underground link example, Hal felt ready to create his own page. He had already written a text file that contained all his contact names—the people at Exemplar and Priority who were on the move committee and responsible for various aspect of the move—so he thought he might just type that into a H.i.P. page for everyone else to see. It would be a basic phone list—nothing complicated. The contact list would be the first H.i.P. page in Hal's first H.i.P. project. So Hal needed a project. Here's what he did:

- Choose **New Project...** in the **File** menu, (or just click on the new page button on the toolbar, 🗋).

A wizard dialog appears. It's the H.i.P. **Template Wizard** , ready to take your order.

- Click on the **New Project** radio button. (Later, you will be encouraged to use the other radio buttons and make use of H.i.P.'s extensive library of templates, but Hal was feeling adventurous.)
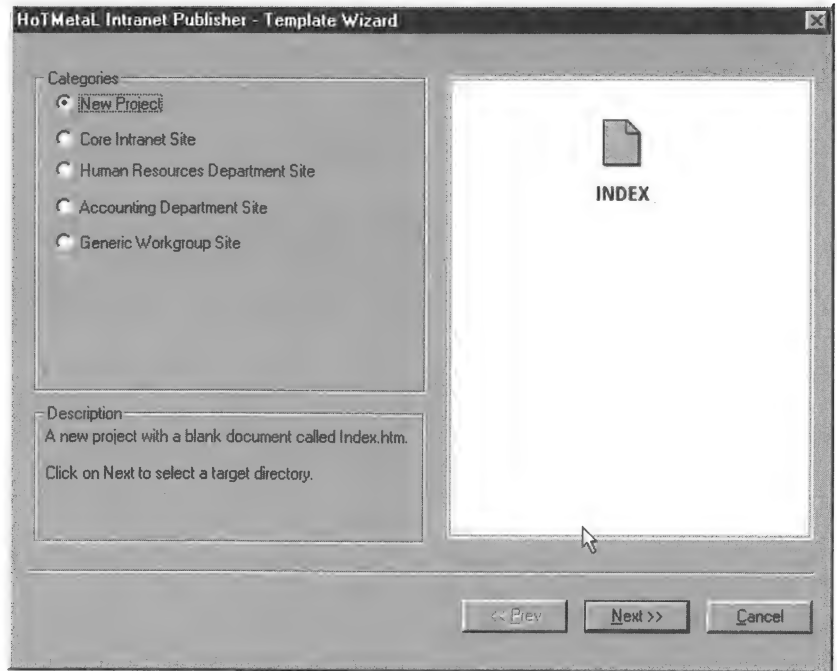
- Click on the [Next>>] button.

The dialog box that appears prompts you to specify a target directory (or 'folder') in which the new project will be created. Name your new folder *the-move* and place it in the *tutorial* folder. When you are finished, the path will be:

```
C:\Program Files\SoftQuad\HiP\Help\tutorial\the-move
```

- Add '\the-move' to the end of the path name in the **Target Directory** text box.
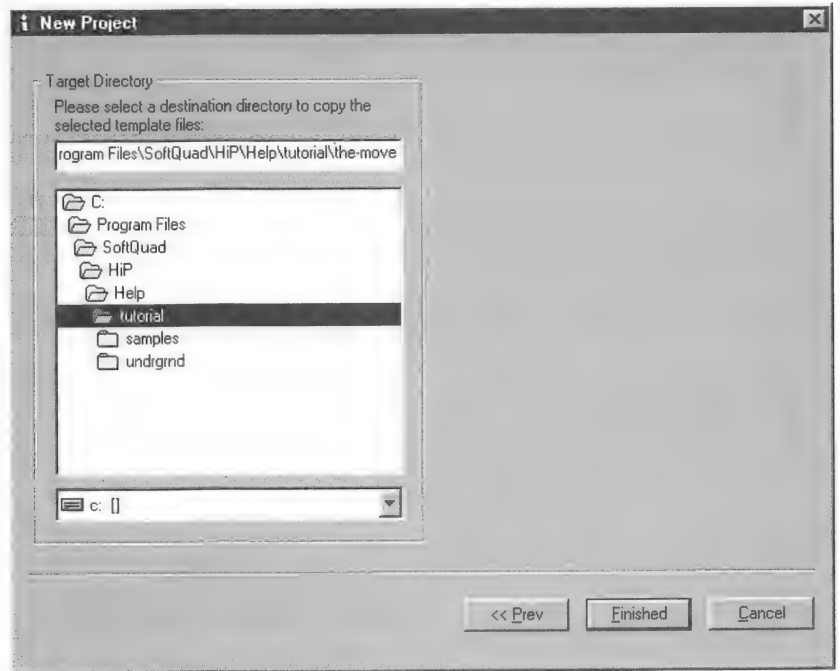
- Click on the [Finished] button. You will be asked to confirm your choice: click on the [Yes] button.

SoftQuad H.i.P. creates the new directory and adds files to it. However, only one of those files is a H.i.P. HTML document—so you will see only one file in the cyberbolic display—it will be called Home Page (if you have **Long Names** chosen in your cyberbolic display). The other files record information about the project, but remain hidden in the Information Manager. There are two files named *project...* and one called *epsilon.hpv*—don't delete them.

Editing the project's first file

While you could start from scratch here and build a file from the ground up, there is almost always a better way. In this case, since Hal has been hard at work on his phone list, we're going to swipe it.

There's a more important reason for using a template or sample file than just getting through a tutorial faster. Making use of existing structures, evolving new ones rather than inventing them, maintaining consistency—these are all important principles for Web site developers in a corporate intranet setting. (By the way, they are fundamental to almost every other publishing endeavor too—think of the consistent structure of many newspapers and magazines.)

Although there won't always be a sample to copy, there will usually be a template with the basic structure in place.

SoftQuad H.i.P.'s advantages really begin to shine when there is a realistic amount of data, so let's import Hal's phone list from the .../tutorial/ samples/ folder.

- Choose **Import File...** (not **Import Project...**) from the **File** menu.
- Click on the [Browse...] button.
- Navigate to the *samples* folder in the *tutorial*.
- Open the file *contacts.htm*.
- Check that your **Destination** is correct: it should be *the-move* folder in the *tutorial* folder.
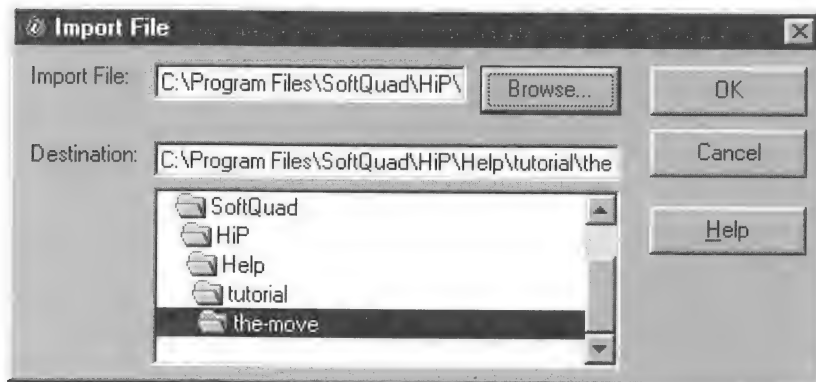
- Click on the [OK] button.

With Hal's phone list imported into 'The Move' project, the Project display has changes: there is a new file *contacts.htm* and there is a new sub-folder called *graphics*. Somewhere in the file *contacts.htm* there are references or 'links' to a a *graphics* folder. Let's take a look. Open the file in the H.i.P. Editor:

- Right click on the file name *contacts.htm* in the Project display.
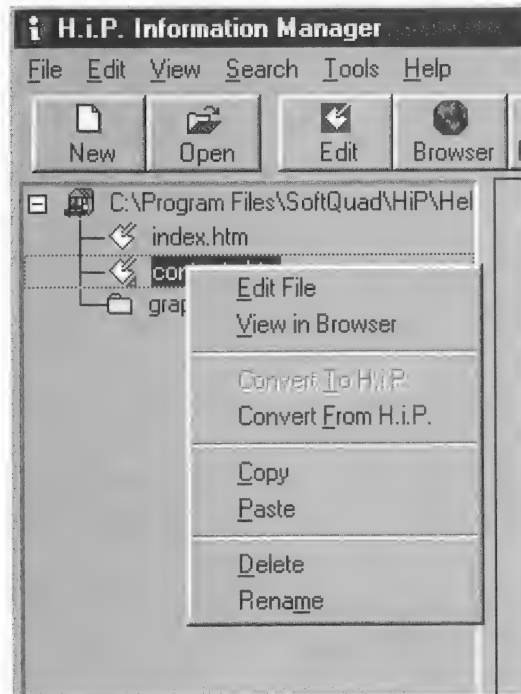
- Choose **Edit File** from the menu.
- Make any small change to the file when it appears in the H.i.P. Editor. (You might correct the line 'Please, not the following' to 'Please note the following'.)

At this point, we would normally just save the file, but let's add on other small task: *contacts.htm* isn't the best name. Let's change it to *phonlist.htm* because this file promises to grow beyond Hal's original concept.

- Choose **Save As...** from the **File** menu.

Did that surprise you? You're looking at a warning message, asking if you want to edit the document's 'properties'.
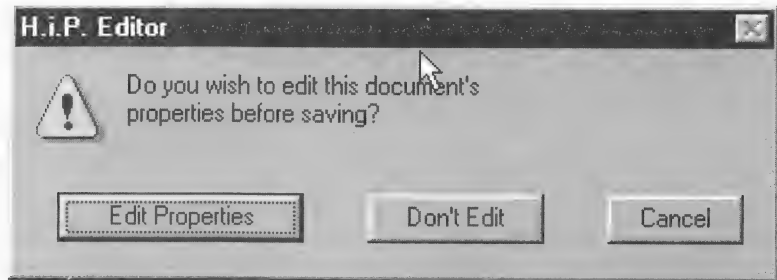
It's a fairly harmless caution, but it's important to find out why it's there. It's time to talk about H.*i*.P.'s helper files. However, we don't want to leave that file without saving it, so just:

- Click on the [Don't Edit] button.
- Save the file as *phonlist.htm* in your project folder, *the-move*, in the *tutorial* folder— but keep it open in the H.*i*.P. Editor.

When you save your file you will notice a short message (with a clock face). Whenever you make a change to project files, the H.*i*.P. Information Manager takes a moment to bring itself up to date by rebuilding its project 'map'.

- Minimize the H.*i*.P. Editor.

Meeting the H.i.P. Helpers

A document with even modest complexity will depend on information stored separately. For example, many intranet documents have associated GIF or JPEG files for displaying pictures. These are 'linked' to the document. A similar approach is used for the H.i.P. features that you may have read about; user-defined extensions (UDEs) are stored in a helper file with the *.hpe* extension, Live TOCs are stored in *.hpl* files, and cascading styles are stored in *.css* files.

Hal has linked a number of helper files to his document. You'll find out soon how to use these and modify them, but right now, we'll have to make sure they are available in your new project.

When a project grows beyond a dozen or so files, you will want to create folders to store groups of files. Usually, these groups are based on the kind of file being stored, and that's the approach we recommend here.

We're going to use the Project display in much the same way that you may have used Windows Explorer to create folders and move files around. First, we have to make a project that includes all of the files and folders we need. This is just like creating *the-move*, but we'll do it one level higher this time and turn the folder *tutorial* into a project folder:

- Choose **New Project...** again in the **File** menu, (or just click on the ⬜ button on the toolbar). The **Template Wizard** reappears.
- Click on the **New Project** radio button.
- Click on the ⌷ Next>> ⌷ button.
- Navigate to the *tutorial* folder, in the *Help* folder, and select it.
- Click on the ⌷ Finished ⌷ button.

Now, all of the tutorial files are available in the Project display and we can move them around by dragging and dropping. We'll get the helper files in place under *the-move* folder first:

- Click on the ⊞ beside the *samples* folder.
- Select the *helpers* folder and drag and drop it on *the-move* folder.

Take a look around your *tutorial* project by clicking on the plus and minus signs. Your *undrgrnd* project is there too.

Now that we've moved the folders into their new locations under *the-move* we're going to reload that project and continue working on it:

- Choose **Open Project...** from the **File** menu (or click on [Open] and navigate to *the-move*.

This time, you will see a file called *project.hpp*:

- Double click on *project.hpp*.

Helper files can be 'hooked up' to the document with a 'link'. That's what we'll do next with Hal's *phonlist.htm*:

- Switch to the H.i.P. Editor—it's probably on your Windows taskbar—and make sure you have *phonlist.htm* open.

- Choose **Properties...** from the **File** menu or click on the [📝] button on the toolbar.

HiP Document Properties   [×]

| General | Effective Date | UDEs | Properties |

⊟ The Move: Contacts
   — Views/Styles
   — Live TOCs
   — Topics
   — Custom META

[Edit...]   [Add...]   [Remove]

[OK]   [Cancel]
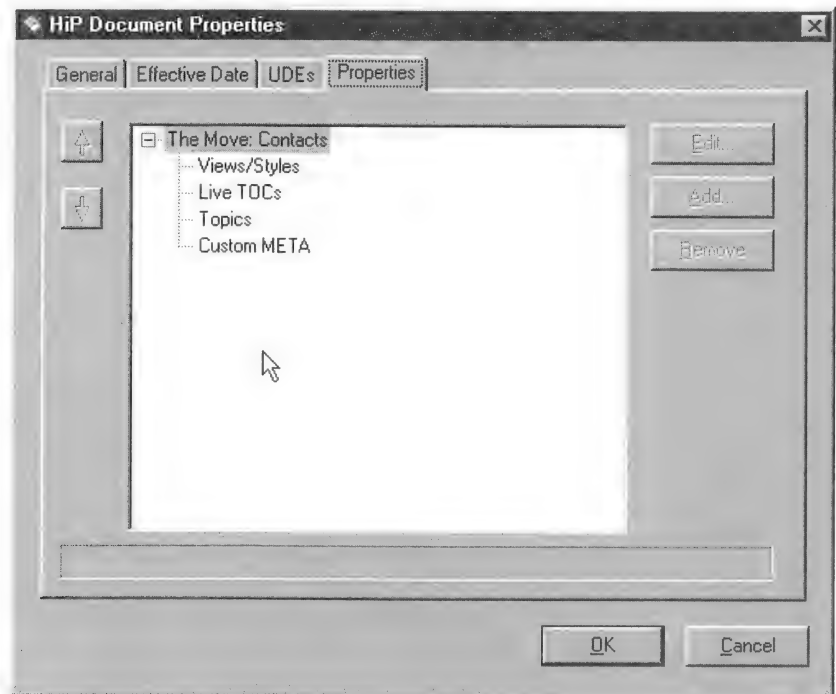
We're going to make use of two of the four tabs in the **Document Proper-ties** dialog: **General** and **Properties**. The other tabs, **Effective Dates** and **UDEs**, will become interesting as the site grows (see page 122).

■   Click on the **General** tab.

Let's give this new phone list a title: how about 'Phone List'? This is the title that will show up in your cyberbolic display—it will be placed in the **TITLE** element of the file.

■   Type `Phone List` in the **Title** text box.

You can add other descriptive information that may be useful, but the real work here is to find Hal's UDE file—that's where Hal has stored a very useful extension that we're going to apply to the phone list.

On a separate tab called **UDEs**, there is a **User-Defined Extensions** section with two buttons: New… , and Browse… .

■   Click on the **UDEs** tab.

We could be creative here and click on New… , get into the UDEs editor and make something totally new! Later. This time, just:

■   Click on Browse… . The **Browse for UDE file** dialog will appear.

A new dialog called **Browse for UDE file** appears.
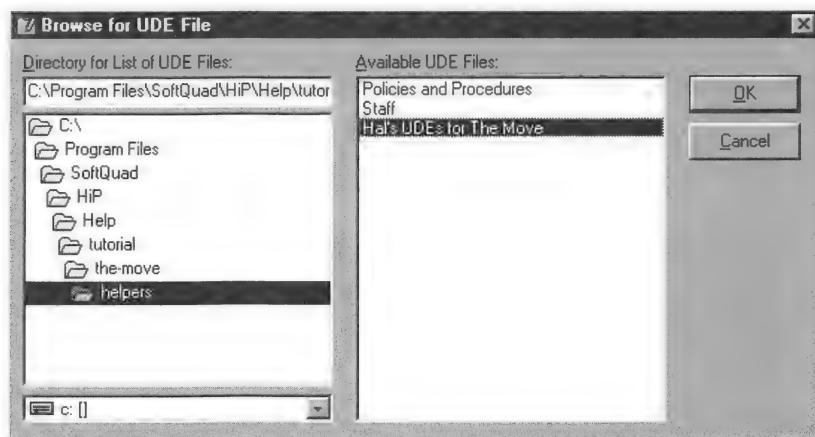
■   Navigate to the *helpers* folder under *the-move*.

- Select **Hal's UDEs for The Move** from the **Available UDE Files** list. Note that these are not filenames, but *titles* of files.
- Click on OK .

A title is now visible on the **Document Properties** dialog: 'Hal's UDEs for The Move'.

One UDE file can store many UDEs, so it's very likely that you and your contributing intranet authors will want to link every document in the project or its sub-projects to one, main UDE file.

Since Hal created some special Live TOCs for his phone list, let's grab those too—even though we haven't really figured out what they are or how to use them.

- Click on the **Properties** tab in the **Document Properties** dialog.

There are four kinds of properties: Views/Styles, Live TOCs, Topics, and Custom Metas. We're going to ignore all but one. (Eager? You can read about the View/Styles property on page 88 and Topics on page 96.) In the meantime:

- Click on **Live TOCs**.

The New... and Add... buttons become active.

- Click on the Add... button.
- Navigate to the *helpers* folder under *the-move* folder.

The Live TOC file we want to link to is titled 'Move Committee Members'. It's this title, not the filename, that is displayed in the **Add Live TOC** dialog:

- Select **Move Committee Members** by clicking in the check box.
- Click on the OK button. Notice that the new Live TOC is displayed in the **Properties** tree.
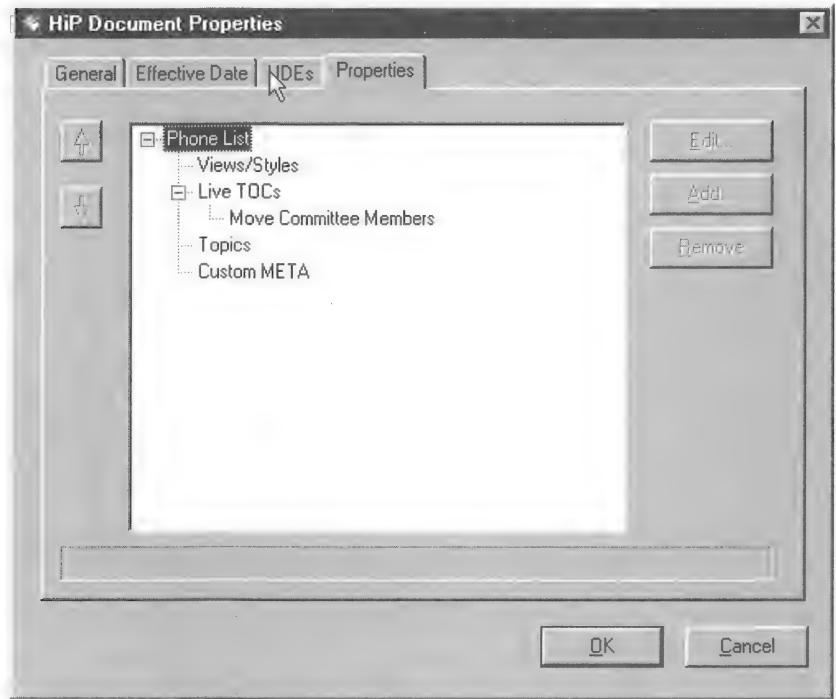
- Click on the [OK] button in the **Document Properties** dialog.

With all of the fundamentals in place, we can do something new with our phone list. At this point, we have:

☐ Created a project

☐ Imported a sample file (Hal's *contacts.htm*).

☐ Linked to a UDE file (Hal's UDEs for The Move).

☐ Linked to a Live TOC file (Hal's Move Committee Members).

But, let's check it out in the H.i.P. Viewer first, just to see how everything fits together.

- Save the file as *phonlist.htm*.

- Choose **Preview...** in the H.i.P. Editor's **File** menu. The **Choose Browser** dialog appears.

You probably added your browsers (Netscape Navigator and Microsoft Internet Explorer) when you installed SoftQuad H.*i*.P. . If they do not appear in your **Choose Browser** dialog, you can add them now. (If they *are* there, just select the one you need and click on [Preview] .)

- Click on the [Add...] button and navigate to your Netscape or Internet Explorer folder.
- Select the program file (*iexplore.exe* or *netscape.exe* for example), and click on the [Open...] button.
- Choose a browser (one that has the H.*i*.P. Viewer plug-ins installed) and click on the [Preview...] button.

Your phone list (with Hal's contact names) is ready to browse, but the appearance of your browser window may seem unusual. You've probably seen Web pages with frames before, so the division of the browser screen into two main panels may not surprise you. However, the buttons at the top of the left frame are definitely H.*i*.P. Before we get down to the work of editing the phone list content, let's take a couple of minutes to play with the functions provided by its H.*i*.P. structure. This is going to be just a brief taste of what H.*i*.P.'s 'Live TOCs' (live tables of contents) and 'Views' can do for a reader; we'll learn how to construct them a little later.

In the left section you can see a list of four names—they're the members of the move committee.

- Click on the **Change Live TOC** icon 📕 .
- Choose **Full Tree** from the menu.

The top line, which says 'Company Directory', has a minus sign, ⊟ , in a box beside it.

- Collapse the list by clicking the minus sign.
- Click the plus sign ⊞ to get the names back in view.

You will see many other multi-level documents where this feature allows much easier understanding of the document's scope and form—imagine a lengthy text that makes use of four or five heading levels.

Phone list names are not normally typed in all at the same time. This is particularly true of Hal's contact list for the move.

The order in which they were typed in is not alphabetical, but we can sort them very easily:

- Make sure all levels are expanded by clicking on all the plus signs.
- Click on the Live TOC ▤ button again, and choose Move Committee Members.
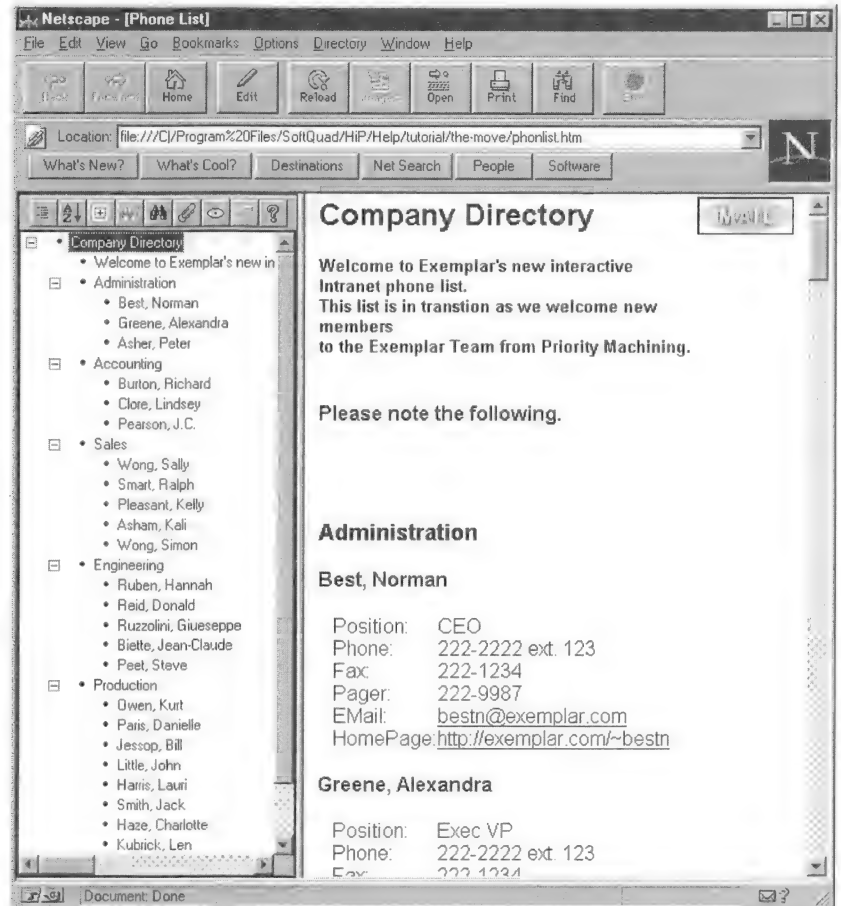
- Click on the ▲↓ button.
- Choose Ascending from the menu that appears.

Now, if there were dozens of names, it would be easier to find one. That's one of the reasons why the TOC is said to be 'live'—because it can act like a table of contents, an index or a small, sortable database. Remember that Hal created his own Live TOC for the move committee members. You can choose it again:

- Click on **Jessop, Bill.**

Bill appears at the top of the right-hand display.

In this phone list, where there are less than 30 names, such a careful organization is probably over-doing it. However, as Hal discovered, data grows. He began with the idea that his simple contact list for the move would always be simple. What he found instead was that his co-workers insisted that it should become a directory for the entire company. With the 400 names that would eventually fill this list, the departmental structure became very useful.

You may have noticed that the **View** icon is grayed out. There are no 'stylesheets' among the properties of this document, and views are controlled by stylesheets. Views are extremely useful so we will learn how to handle them later in the tutorial (see page 111). In the meantime, we'll take a look at UDEs and how to apply them to a document.

By the way, you have a file in your project that doesn't have to be there: *contacts.htm*. You can delete it using the H.i.P. Information Manager just like you would in Windows Explorer:

- Switch to the H.i.P. Information Manager.

- Right click on *contacts.htm* and choose **Delete** from the menu.

- Confirm the deletion by clicking on Yes .

While you are here in the Project display of the H.i.P. Information Manager, you may want to make sure that all of your documents are H.i.P. . Any that are accompanied by an icon like this are ordinary HTML. To convert them:

- Right-click on the file to be converted, and choose **Convert to H.i.P.** from the menu.

Working with UDEs

At this stage, it's a little early to start making UDEs of our own—we'll do that later (see page 75). Luckily, Hal has already done some ground work for us. What we're going to do now is take advantage of his UDE definitions to designate the names and phone numbers of a couple new members of the move management committee. Hal has created a user-defined extension (a UDE) called 'CommitteeMember' that we're going to use—it's in 'Hal's UDEs for The Move'—that's the title of the UDE file that we linked into our phone list.

So let's go ahead and appoint some new members to the move committee by finding and marking them up with the CommitteeMember UDE:

Normally, you will be working with three programs when you edit H.i.P. documents: The H.i.P. Information Manager, the Editor and your H.i.P. Viewer. (You can also choose settings for the Monitor Administrator through its Web page interface in your browser, but that happens less often.) You will be switching between programs frequently—in fact, there's hardly anything you will learn to do in this tutorial that doesn't require all three programs. Keep in mind that the H.i.P. Information Manager is the 'control tower' and it should always be your starting point.

- Select *phonlist.htm* in the Project display of the H.i.P. Information Manager.

- Click on the H.i.P. Editor button ▦ .

After we have surrounded two new committee members' names in CommitteeMember UDEs, they will also appear in the Live TOC display.

First, we are going to find 'Anthony LaSalle'.

- Choose **Find and Replace...** from the **Edit** menu, or click on the ▦ toolbar button, or type (Ctrl-F) .

- Type LaSalle in the **Find** text box. You don't have to use **Match Case.**

- Click on (Find) .

Anthony's information is contained within a (TABLE⟩ element.

- Highlight the entire TABLE element by clicking on the (TABLE⟩ start tag with the left mouse button.

- Choose **Insert Element...** from the **Markup** menu, or click on the tool-bar button, or type Ctrl-I at the keyboard.

The **Insert Element** dialog appears, listing both standard HTML elements and Hal's UDEs.

- Select 'CommitteeMember' and click on OK .

This surrounds the selected information with the CommitteeMember tag.

You have just made Anthony a member of the move management committee—at least as far as the phone list is concerned. Take another minute to put Kali Asham on the committee too (she's above Anthony, so you might do a **Backwards Search** for 'Kali' or 'Asham'.).

With both Kali and Anthony designated as members of the move committee, their names can be presented as two new lines in a Live TOC. Hal has already created this particular Live TOC—we will create our own later (see page 83).

Go to the H.i.P. Viewer now to see where Kali and Anthony are listed. (Actually, there are two other ways to do this: via the H.i.P. Information Manager or by loading the file into the H.i.P. Viewer):

- Choose **Preview...** from the **File** menu, and follow the instructions in the dialog to choose your browser application.

Or:

- Click on the taskbar button to activate the H.i.P. Information Manager, select the file, then click on the H.i.P. Viewer button .

When the H.i.P. Viewer has finished loading the new version of the phone list:

- Click on the **Change Live TOC** button, and choose **Move Committee Members**.

Both 'Asham, Kali' and 'LaSalle, Anthony' are there in the committee list.
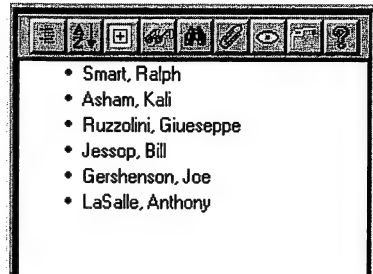
Their information is displayed in the right panel when you click on their names in the Live TOC. You can add other names to the move committee by surrounding their information with a **CommitteeMember** UDE, or you could create any other special grouping and give it its own UDE. Notice that Kali's and Anthony's names are still displayed in the regular way in the default Live TOC. Kali is in Sales and Anthony is in the production regardless of their special status as members of the moving management committee.

- Click on the Live TOC button, ▥.
- Choose **Full Tree.**
- Check to see that Kali and Anthony are listed by expanding the levels.

Incidentally, when the move is complete and the move management committee is disbanded, Hal will be able to remove the special identification from the former committee members' names just by selecting the information within the **CommitteeMember** element and removing the tag.

- Switch to the H.i.P. Editor.
- Select the (TABLE⟩ that contains Kali Asham's information (without selecting the surrounding **CommitteeMember** tags).
- Click on the 🞩 button on the toolbar.

That's how it's done. But, since you probably want to keep Kali on the committee, you should undo your changes:

- Type (Ctrl-Z), or choose **Undo** on the **Edit** menu.

At this point you may want to take a breather. You have already been exposed to some of the more advanced features of H.*i*.P.—opening projects, editing files, and linking to helper files. You may find that editing H.*i*.P. HTML files is a new set of skills. If this is the case, you might stop here and take some time to develop your HTML editing skills. There is a tutorial chapter just for this purpose—see page 135.

Creating additional pages and simple links

Sally Wong was so excited about having a corporate intranet that she had already submitted several pages of material hoping that Hal would publish it immediately. Sally runs the sales office for Exemplar. Hal appreciated the enthusiasm and he was anxious not to disappoint anyone. The problem was that Sally's office works in WordPerfect and they had never heard of an HTML document. Hal explained about Hypertext Markup Language but wasn't able to make much of an impression on Sally; she wanted to see it on her screen right now. Ah ... sales people!

Converting a word-processor file

Hal decided that, at the very worst, he could just retype the document. But, as he was about to create a new file he wondered ... could a WordPerfect file be converted automatically to HTML? Instead of clicking on the **New** tool bat button ▣, he decided to poke around a little. In the **Tools** menu, he found **Convert Documents**.... "That," Hal smiled to himself, "would be the cat's pajamas." Here's how to do it:

- Switch to the H.*i*.P. Information Manager.
- Choose **Convert Documents...** from the **Tools** menu. The **Select Files to Convert** dialog appears.
- Navigate to the *samples* folder under *tutorial*.
- Expand the **Files of type** drop down list at the bottom of the dialog.
- Choose **WordPerfect [*.wpd]**.
- Select *slspack.wpd*.
- Click on the [Open] button. The **Choose Target Folder** dialog appears.

You can leave the converted file in the *samples* folder.

- Click on the [OK] button.

After a few moments you will see **Batch Document Conversion** message with a status bar. Then a message like this:

Batch Document Conversion [×]

Batch conversion complete.

Please check C:\Program Files\SoftQuad\HiP\tutorial\samples\SQcnvDoc.log for details.

[OK]
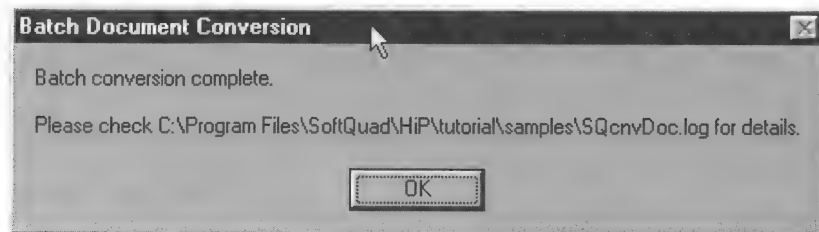
If you would like, you can read the conversion log file.

You will find a new file name in the *samples* folder, called *slspack0.htm*.

■ Import the file from the *samples* folder (where the conversion process placed it) into your project using **Import File...** on the **File** menu.

Once the newly-converted file is shown in your Project display:

■ Right-click on *slspack0.htm* and choose **Rename** from the menu.

■ Place your cursor in the white text box and edit the name. Change it from *slspack0.htm* to *slspack.htm*.

■ Type (Enter).

■ Convert the file to H.i.P. (Right-click again and choose **Convert to H.i.P.**)

Can it be that easy?

"Well, yes and no", Hal said to himself. It's easy because the H.i.P. Editor converts the WordPerfect document (or Word, Rich Text Format, etc.) into good HTML. It's not easy because there is no automatic way to structure a document that is just one lump of text. Fortunately, Sally has been fairly consistent with her subheads and the conversion has picked up on these and turned them into tagged elements. Of course, it has also marked up each paragraph. But, in order to present the document in a way that would be easy to navigate, Hal has to do some hand work.

To complete the document for H.i.P. viewing:

■ Highlight the file *slspack.htm*

■ Click on the [Edit] button.

H.i.P. Editor - [slspack.htm]

File   Edit   View   Markup   Format   Tools   Forms   Special   Window   Help

HTML > BODY > H1 **Packing Instructions for**

BR > /BR

Exemplar and Priority Sales Personnel /H1

P > Each sales representative is responsible for packing his or her own files, books, catalogs and other business materials. The following memo describes how these items should be packed and identified so that the moving contractors can do their job efficiently. /P

H2 > What to pack /H2

P > All of your own business files and materials will have to be packed; basically, that's anything on or in your desk and your shelves. This also includes your computer keyboard and mouse. /P

H2 > What not to pack /H2

P > You should not pack your computer (the box) or your computer monitor. These will be prepared for the move by the systems staff. /P

P

Rules Checking: On

As he was typing a title in the Document Properties dialog, selecting and surrounding text, inserting and filling various markup elements, Hal reflected on how much effort could be saved if documents were all created with a machine-readable description of their structure. Then, he mused, you could send them anywhere, and anybody's computer could

present them properly just by automatically examining a structural plan and putting together the pieces. "Hmmm... I wonder if anybody has considered that?"

- Click on the ![button] button on the toolbar.
- Save your file if **Save** dialog appears.

When Hal took a look at his work in the H.*i*.P. Viewer and was surprised. He didn't really expect to see a table of contents just appear automatically. But there is was.
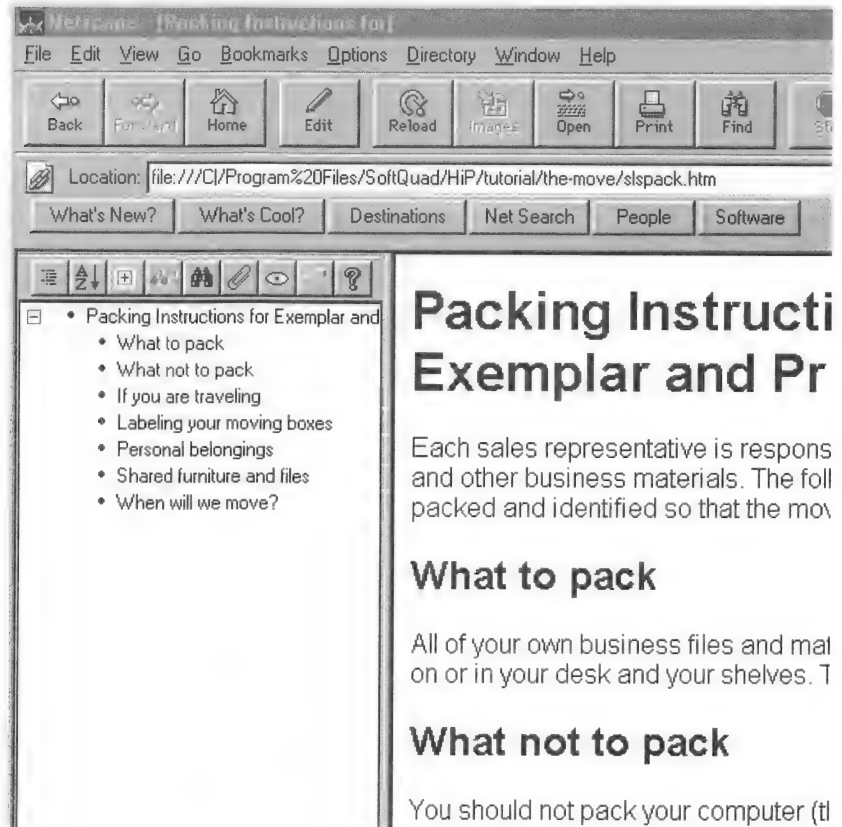
Live TOCs are created whenever a H.i.P. document is viewed. As long as the document has some heading elements (H1, H2, H3, etc.), the H.i.P. Viewer takes care of the rest automatically. Hal clicked on the collapse and expand boxes just to reassure himself that they worked. Sally the sales manager would be impressed.

Of course, even more useful Live TOCs can be created, referring to any element you like—including elements that you define (UDEs). If you'd like to learn about custom Live TOCs right away, see page 83.

While the Sales Office Packing Instructions were now in H.i.P. format and had their own (default) Live TOC, Hal knew that it wasn't over yet. There was still a lot that he could do to make the document attractive and useful—including the creation of different views of the document for different people. Even though this was in Hal's future, you can jump ahead (see page 88).

Working with straight HTML

Priority Machining Ltd., the company that Exemplar Manufacturing had just bought, already had a Web page, so their systems people were familiar with HTML But, when he asked the Priority people to send in their information about the move, he found that it was not the first thing on their list. A good example might help them identify their priority, he thought, so he decided to e-mail Sally's page.

After his experience with conversion from word-processor files, Hal was fairly confident that he could take an ordinary HTML file and turn it into H.i.P. But he wasn't sure how to convert H.i.P. files for use by non-H.i.P. Browsers. He began by copying *slspack.htm* to a new file name:

- Switch to the H.i.P. Information Manager.
- Right-click on *slspack.htm* in the Project display, and choose Edit File from the menu.
- Choose Save As... from the File menu and save the file under a new name: *example.htm* in *the-move* folder. (You don't need to edit the document properties—just click on the ⎡ Don't Edit ⎤ button.)

There is a new file in your Project: *example.htm*.

Once Hal had his new file, he converted it to HTML and sent it off to Priority:

- Right-click on *example.htm* and choose Convert from H.i.P..

Notice that the file's icon has changed in the Project display.

Of course, no one had any difficulty browsing the file because it was standard HTML. On the other hand, no one had a chance to see it as a H.i.P. document either. The solution to that problem, Hal discovered, was to give them all a copy of the H.i.P. Plug-ins. In fact, if you try to browse a H.i.P. document without the Plug-ins, you will be connected to SoftQuad's Web site so that you can download and install them.

The HTML documents that Hal eventually received from the Priority people were easy to handle. Hal was eager to add some H.i.P. capability to the Priority page, just to show them what could be done, but he was relieved to learn that, if necessary, the pages can be functional as ordinary HTML with only a simple conversion.

Links that ride off in all directions

Now that he looked more closely at the ordinary HTML pages that he had received from the Priority people, Hal decided there was a H.i.P. improvement he could make. Like many other Web pages, these ones contained a long section of links. There must have been a couple of dozen of them and there was nothing to differentiate or classify them. "Time for a multiple link menu," Hal muttered as he opened his H.i.P. Editor.

Hal had already had a look at a multiple link in its final form so he had an idea what he was getting into. If you haven't seen, we'll go back the the 'Underground' project that you opened earlier in the tutorial:

- Switch to the H.i.P. Information Manager if you aren't already using it.

- Choose **Open Project...** from the **File** menu, and choose the *project.hpp* file that's in the *undrgrnd* folder in the *tutorial* folder.

- Right-click on the homepage (it may be labeled 'Oxford Circus' or *index.htm* depending on the cyberbolic display option you have chosen).

- Choose **View in Browser**.

- Scroll toward the bottom until you see this button: ▣.
- Click on the button
- Click on one of the station names.

A menu of links appears. A menu of links is also called a 'one-to-many' link since it links a single point to many other locations, or a 'multiple location' (multiloc) link .

Most of the menu items on the first level lead directly to a page. But it's also possible to have an item on the first level lead to a second level, and on to a third. An example of that is the menu item **Bond Street**. Click on it and see where is leads. (In case you are tempted by **Marble Arch**, don't forget, it's a broken link.)

With such 'nesting' of menus and submenus, you can stack a lot of links in a very small space. You can also group them more logically and label the groups. The rule separator that you see between 'Green Park' and 'Bond Street' is easy to make, as you will see.

Here's how Hal began to organize the Priority material. First, he imported the Priority files into his project:

- Switch to the H.i.P. Information Manager.
- Open 'The Move' project (**Open Project...** on the **File**).
- Choose **Import File...** from the **File** menu. The **Import File** dialog will appear.
- Click on the ⟨ Browse... ⟩ button.
- Navigate to the *samples* folder in the *tutorial* folder.

Now, here's a good spot to stop and take a look around. The file that you are going to work on is *p-cat.htm*. You could import it by itself. But, there's a better way to do this. Since the Priority files are a group, and since that group is linked together by the Priority Home Page file, *p-idx.htm*, it will be much easier to import that one.

If you were to import *p-cat.htm* on its own, you would eventually have to reconnect the Priority links when you went back later to import the other four files. If you import *p-idx.htm*, it will not only bring the other files along with it, but the whole group will arrive with all of the links rewritten to reflect their new positions within *the-move*.

- Select the file *p-idx.htm*.

- Check to make sure that the **Destination** is correct (it should be *the-move*).
- Click the OK button.

When the files are all imported (there are five), your Cyberbolic display will be overflowing. However, some files in the project are still not linked to anything. You can verify this:

- Choose **Find Orphan Files** in the **Search** menu.

Remember that mystery panel that we didn't talk about? It's called the H.*i.*P. Pocket display. You can group files together in pockets and collect them here in several ways. Searching for orphan files is just one example—it creates a H.*i.*P. Pocket called **Orphan Files**, then deleted it.

- Click on the ▓ to expand and collapse it.
- Right click on **Orphan Files**.
- Choose **Delete H.i.P. Pocket** from the menu.

Now we can go ahead and make our changes to the file *p-cat.htm*.

- Click your right mouse button on *p-cat.htm* in the Project display, and choose **Edit File** from the menu.

Hal was eager to organize the links starting with 'High-Pressure Forged Products' down to 'High-Polish Aluminum'. As a first grouping, he decided to choose the first four. But, since all of these links were surrounded by a UL, he just changed the UL to a Multiloc. We're going to follow his method:

- Put the insertion point immediately to the right of the first (UL⟩ start-tag.
- Type Ctrl-L or choose **Change Element** from the **Markup** menu.
- Choose **Multiloc** from the **Change** dialog box.
- Click on OK .

When you are done, your document will look like this:
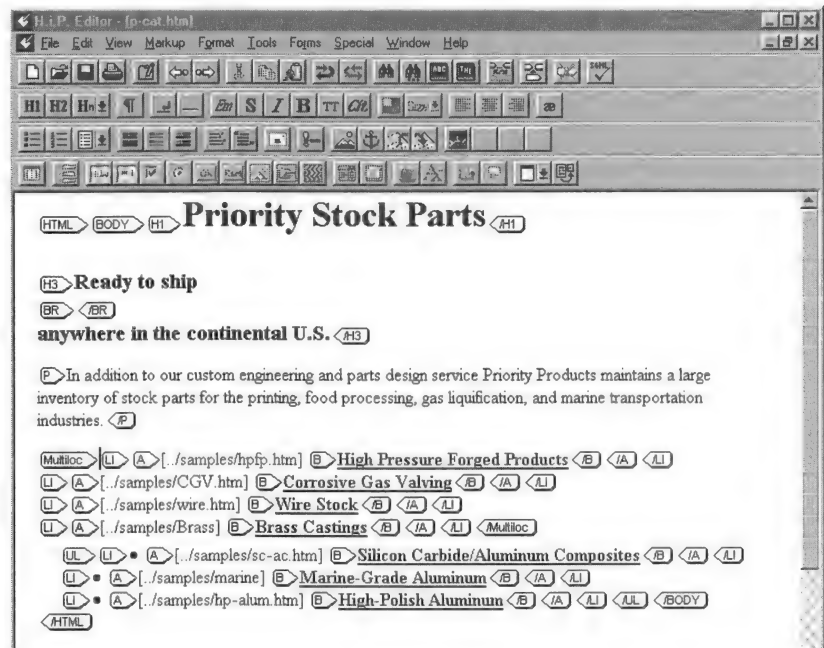
With these steps, Hal collected a group of links (anchor elements) in the document, and designated them as a special kind of H.i.P. element—a 'multiloc' or multiple location link. H.i.P. takes care of the rest. Take a look at your work in the H.i.P. Viewer.

- Save the file.

- Choose **Preview...** from the **File** menu, click on your viewer toolbar button, or type ⟨Ctrl-M⟩.

When your H.i.P. Viewer displays the page, you will not see the links that you surrounded. In their place will be a button:

- Click on the ⊞ button.

The anchor text that you surrounded with a Multiloc element is now contained in a pop-up menu.

The situation is a little different if you are adding new text. In that case:

- Switch to your H.*i*.P. Editor and the file *p-cat.htm*.
- Place your insertion point just before the ⟨/BODY⟩ tag (this is where your new multiloc will appear).
- Click on the ▦ multiloc toolbar button.
- Select the LI element—the one that's inside the new Multiloc⟩ .
- Type Ctrl-C to copy it to the Clipboard.
- Place your insertion point just to the right of the Multiloc⟩ start-tag.
- Type Ctrl-V twice to make a total of three A elements within three LI elements.

Now we're ready to invent three new links. (But, we won't bother to create any files for these to link to.) Let's give it a title.

- Place your insertion point just before your new Multiloc, (the second of the two).
- Type Your reliable Priority Representatives. A new P element is added automatically.
- Select the content of this new paragraph and type Ctrl-L , then choose H4.
- Place your insertion point within the first A element.
- Click on the ⚓ .
- Place your insertion point in the text box labeled Path.
- Type slsbrass.htm and click on OK .
- Type Brass casting sales reps at the insertion point (inside the A element).
- Repeat the process in the other two anchors using:
  - □ slsmar.htm and Marine metals sales
  - □ slshpv.htm and High pressure valving sales

Here's how your editing screen will look:
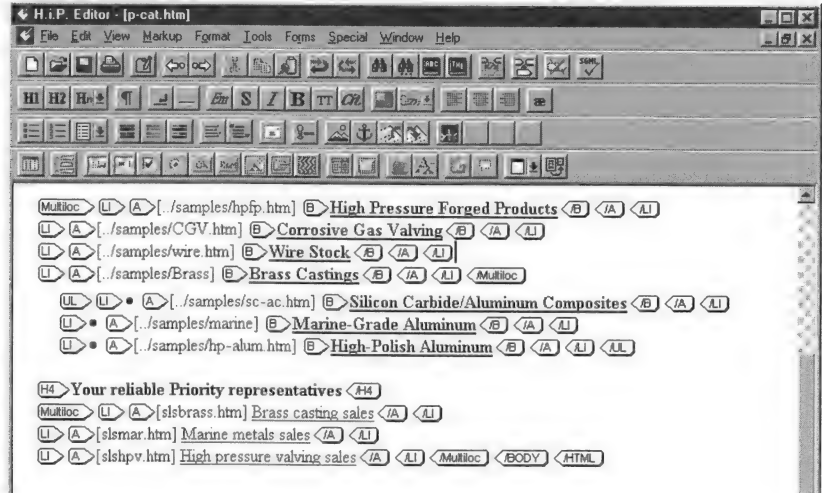
- Save the file and preview it.

Of course, if this were a real page, not just an exercise, you would make links to new pages with pictures and contact information on you sales reps.

One last refinement. Add a horizontal rule to separate 'Brass Castings' from 'Wire Stock':

- Place your cursor just after the opening tag `LI` for 'Brass Castings'.

- Press `Ctrl-P` to get a new LI element. Leave it empty.

- Preview your document and click, again, on the first ▶ multiloc button .

- Close your file and minimize the H.i.P. Editor.

If you're enjoying this, you can go back the H.i.P. Editor and create some nested menus as described in the next section. However, if working in a structured document is a new experience for you, you might be feeling overwhelmed. In that case, you should skip the next section and resume

reading with *Creating a root page* (see page 69). There is also a tutorial chapter that can help you acquire some new HTML editing skills—see page 135.

Ready for some nested multi-location links?

You have already created a simple, un-nested multiple-location link by placing a group of LI elements that contain A elements within a Multiloc. Congratulations! You are still in the early stages of the tutorial and you are already using one of H.i.P.'s more sophisticated features.

Nesting just means placing menus within menus—within menus. It may start to look complicated on the screen, but that's all it is. Let's see how it's done:

■ Switch to the H.i.P. Editor with the *p-cat.htm* file open.

■ Place your cursor just before the last closing ⟨ /LI ⟩ tag within the Multiloc (its the one that contains 'Brass Castings'.

■ Press ⟨ Ctrl-P ⟩ . This provides another 'menu item' in which you will place a whole Menu group of LIs.

■ Type `Aluminum Products` into that last LI, but don't surround it with an A element as has been done with the other list items.

■ Put the insertion point just after the ⟨ UL ⟩ start-tag.

■ Type ⟨ Ctrl-L ⟩ and change the element type from UL to Multiloc.

Now you have the aluminum products enclosed within their own Multiloc.

■ Select the whole content of the new Multiloc element by clicking on the opening ⟨ Multiloc ⟩ start-tag.

■ Drag your selection by left clicking anywhere within it, then moving your cursor to the point just before the ⟨ /LI ⟩ end-tag of the list item containing 'Aluminum Products'.

■ Release the mouse button to drop the selection in its new location.
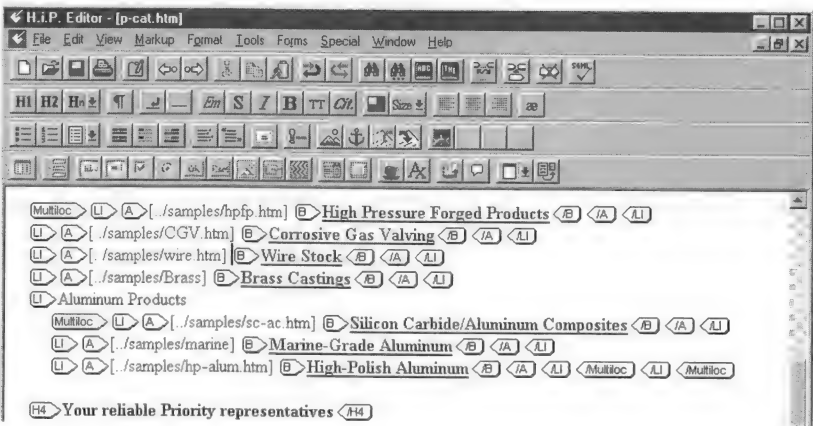
This is how the whole thing will look on your H.i.P. Editor screen. And here's how it looks in the H.i.P. Viewer.
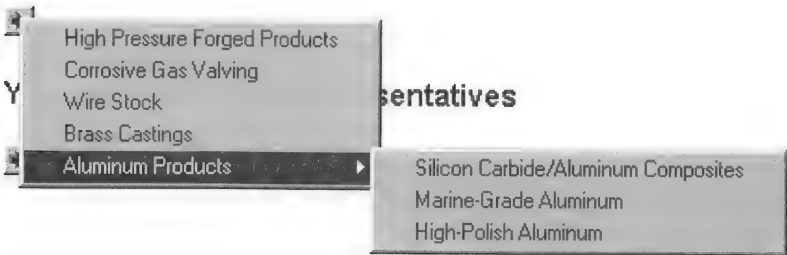
Good work! Switch to the H.i.P. Editor, save your file and close it.

Now a confession: the links you have just placed in your nested multi-location link menu aren't connectable to anything. When you finally examine your project in the cyberbolic display you are going to 'see red'. But before that can happen, we will need a 'root' page and that's in the next section.

After creating his first multiple-location link menu for the Priority pages, Hal is inspired with an ambitious thought. In his H.i.P. Editor, he had noticed the [image] button for inserting images, and was wondering whether the Priority pages would work better if there were some pictures of the products. But then he had a more immediate concern: he'd never met his

contacts at Priority and he wondered what they looked like. The Priority employees who were about to join Exemplar through the acquisition would be unfamiliar to the rest of the company, and vice versa. Pictures would make it a lot easier for everyone to get acquainted. Hal decided on a series of portrait pages.

But it was a big job—Hal needed some help. He asked his colleague Aleksandr to collect employee photos and scan them as *.gif* files, then to create pages that consist of just one picture. They agreed on filenames and Aleksandr installed a copy of the H.i.P. Information Manager, H.i.P. Editor and H.i.P. Viewer. Finally Hal was beginning to feel like a publisher.

Creating a 'root', or 'home' page

Hal realized that he was lacking an important part of his project. The pages that he created or converted so far—the phone list, the Priority pages with all their links, and the sales office packing instruction page— none of these had any particular relationship to each other, beyond the fact that they were all in the same project. Hal felt he needed a place to start: something to serve as the root of the tree of pages that was beginning to grow.

In a paper document, this purpose would be served by the title and contents pages, but on the intranet, this 'root' page is known as the 'index' and is usually called *index.htm*. When Hal created his new project, *themove*, H.i.P. created a file called *index.htm*, "without even being asked", Hal muttered to himself. He felt mildly resentful, but there it was, he thought, "I might as well make use of it."

To open the file *index.htm* in your H.i.P. Editor:

- Right-click on the document 'Home Page' in the cyberbolic display (or, if you have file names displayed, right-click on the file *index.htm*). Note that you want your original Home Page for the project, not the Priority Home Page (*p-idx.htm)* that you imported.

- Choose **Edit File** from the menu.

When *index.htm* appeared in his H.i.P. Editor, Hal spirits lifted. It was blank. A clean slate. Hal began writing the text and adding **A** elements to contain his links to other pages.

However, at this point, we'll save some typing by swiping Hal's work again. We'll open up a sample document and save it as our own file called *index.htm*. Let's close the *index.htm* file first. There's no need to save it.

- Choose **Close** from the **File** menu.
- Choose **Open...** from the **File** menu.
- Select the file *halindex.htm* from the *samples* folder in the *tutorial* folder.
- Click on the [Open] button.

Now we get a chance to add a title to our new home page.

- Click on the [🖉] document properties button on the toolbar.
- Type The Move Home Page in the **Title** text box of the **General** tab of the **Document Properties** dialog.
- Click on the [OK] button.
- Save the file under the file name *index.htm* in the *the-move* folder (overwrite the old *index.htm*).

The H.i.P. Information Manager takes few seconds to trace outwards along the links, starting from the root page and extending all the way out to the **mailto** addresses of the people in the phone list.

Hal right-clicked on the cyberbolic display and chose **Show Filenames** from the menu. That's when he discovered that there was a link to a file that wasn't yet in the project. The rest of the Priority pages are in samples. So let's go get them.

If you recall the way we moved the *graphics* and *helpers* folders into *the-move* project folder, you might be tempted to try the same method to move a group of documents. This is what Hal did: he opened the *tutorial* project, then dragged and dropped files from the *samples* folder to *the-move* folder.

Not a good idea. Having done it that way, Hal would *not* recommend it. The files ended up in the right place—no problem there, but Hal had to then reconnect the links one-by-one. When files are dragged and dropped within a project, the links are maintained. However, when Hal reopened *the-move*, his links were external to that project (they were displayed in blue) because they were still aimed at the old copies back in the *samples*

folder. He had to change them, in the H.*i*.P. Editor, to connect to the new copies that he had made in *the-move* folder. Tedious!

The answer is to let the H.*i*.P. Information Manager do the work. "*Manager do work*. There's a concept," Hal mumbled.

When you are importing groups of linked files, you may find that there is a 'file name clash' with something that's already in your project. For example, let's say that you had imported *p-cat.htm* by itself, did some work on it, then went back to the *samples* folder and began to import *p-idx.htm*—which has a link to an old version of *p-cat.htm* in the *samples* folder. Here's what would have happened when the link (in *p-dix.htm*) to the old version of *p-cat.htm* was detected:
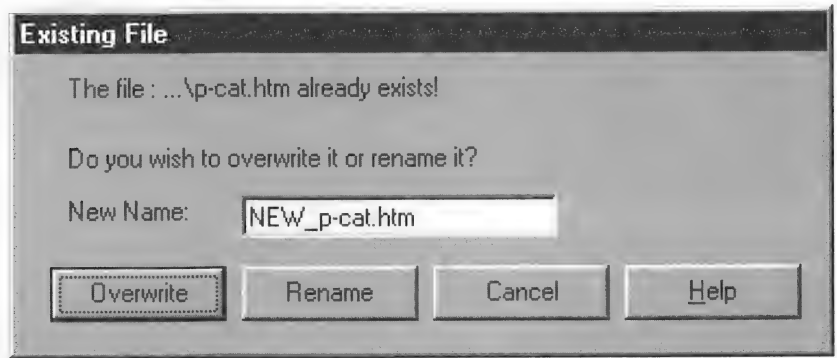
Since *p-idx.htm* contains a link to *p-cat.htm*, H.*i*.P. tries to make sure it's imported. In this case, we already have it (and we've made edits to our version) so, normally, we wouldn't want to overwrite it. If this happens, and you want to avoid overwriting your newer version:

- Click on the [Cancel] button, and confirm that you want to cancel the operation by clicking on the [Yes] button.

The operation that is cancelled is the overwriting of the file—the rest of the import process proceeds.

By the way, if you need a little elbow room you can enlarge the Project panel:

- Place your cursor on the center of the border at the edge of the Project panel.

- When your cursor changes to a vertical bar with opposing arrows, drag the border to a more convenient position.

As each file is imported, it appears in the Project display as part of the tree structure. However, the cyberbolic display doesn't really change. Even though there are links within the imported documents, they haven't yet been displayed. If you'd like to see them—you know you would—then remap the display:

- Click on 🔄 Remap .

Your cyberbolic display will look something like this:
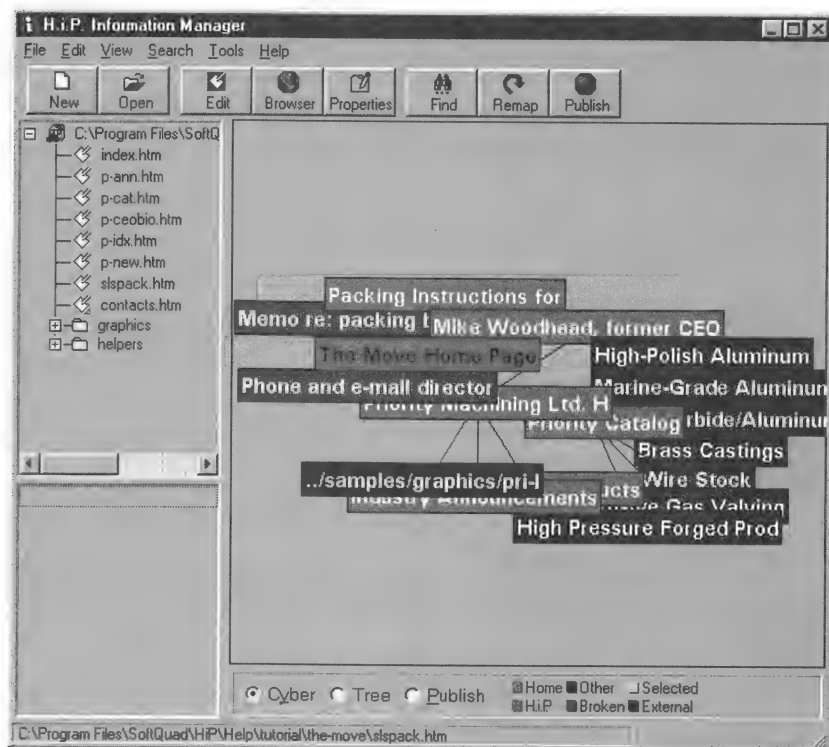
If you have any files that need to be converted to H.*i*.P., now is a good time:

- Select the non-H.*i*.P. files—the ones with the icon.

- Right-click on your selection (there can be more than one file), and choose **Convert to H.i.P.** from the menu.

Hal was delighted with the image his project made in the cyberbolic display and he spent a few minutes just pulling the labels around the screen.

After he viewed his home page again (this time he just clicked on the label with his right mouse button to launch the H.*i*.P. Viewer), it occurred to Hal that this would be a good place to learn more about cascading style sheets because his home page was a little plain. Well yes, a home page should have style, but there were some more basic skills to learn first.

In the meantime, he was going to have to figure out some aids to navigation for his soon-to-be intranet readers. "Gee whiz," thought Hal, "if I had a tutorial to learn all this stuff, I'd already be reading Chapter Two!"

Helping readers find their way

Finding your way around a few simple, familiar pages is easy (and you don't need a H.i.P. Viewer to do it). But good, active intranets grow like mushroom colonies. Getting what you want out of large collections of text and pictures that are unfamiliar and constantly changing—getting the right mushroom—takes some special navigational intelligence. Now that Hal's moving project has begun to acquire some mass, we can show you a few H.i.P. tricks that will save everybody a lot of time.

Adding to the H.i.P. HTML tag set with user-defined extensions	Hal's first experience with UDEs was making the CommitteeMember UDE that you used to place Anthony LaSalle and Kali Asham on the move committee back in the first chapter (page 53). After that, Hal was itching to try it again. He wasn't sure what else a UDE might be good for, but they sounded so important...so H.i.P.

The idea is that you can work within the confines of 'standard' HTML— stick with the tried, true and universally accepted bunch of elements. But, you can also increase the variety of elements at your disposal by extending the standards ones, using their CLASS attributes. If you want to learn more about elements and attributes, see the chapters *Core* HTML and *Extensions to* HTML in the H.i.P. *Editor* section of the *Reference Guide*.

To be sure, the extended elements perform their special magic only if the reader is using the H.i.P. Viewer, but the H.i.P. Plug-ins are available from SoftQuad's home page: *http://www.softquad.com*.

Before Hal chose an HTML element on which to base his **CommitteeMember** UDE, he considered what ordinary HTML elements really are. A few HTML elements just describe what things should *look* like in the viewer: **B** (bold text) is one of those. But the elements that really make things hum are the ones that define what their content *is used for*. Titles are a good example: the title element is displayed in the title bar of the browser, but it is also used by the browser to keep track of where the user has been—the content of the title elements is also collected in a list on the **Go** menu in your browser. Very useful!

Hal was looking for a new problem to solve with a user-defined extension—something that would be useful because it was a particular kind of *thing*—not because it needed a particular kind of *look*. And then he knew he had one.

There were many, many items to be moved that would require some special treatment. This was especially true of production equipment that would need lots of preparation by the plant staff, but it was also true of many pieces of office equipment, and even the office plants. Hal realized that he could create a new kind of element that could be used just like any other HTML element—a new user-defined extension that could be applied to all of the special moving-related tasks and problems. He would call his new tag 'SpecialAttention'.

Just before we see how he created his UDE, we should point out that spaces between words in a UDE name are not allowed. Use a separator, like '-' to take the place of spaces. Hal's 'mixed case' approach—joining the words but keeping them capitalized, is a useful convention for distinguishing user-defined extensions from the standard HTML elements.

Here's how Hal created 'SpecialAttention':

- Switch to the H.*i*.P. Information Manager.
- Choose **User-Defined Extensions...** in the **Tools** menu. (UDEs are created in the UDE editor which is accessible from either the H.*i*.P. Information Manager or the H.*i*.P. Editor.)
- Open the file *the-move.hpe*—it's in the *helpers* folder in *the-move* folder. (UDEs are collected and stored in *.hpe* files). When you need one or more UDEs for a particular document, you add them through the **Document Properties** dialog.

Once you open the *.hpe* file in the **UDE Editor** you are presented with the **H.i.P. User-Defined Extensions** dialog. There's lot's going on in this dialog, so take a few minutes to have a look.

```
╔══════════════════════════════════════════════════════════════╗
║ ⚿ HiP: User-Defined Extensions                    _ □ ✕ ║
╠══════════════════════════════════════════════════════════════╣
║          Extensions List Name:  Hal's UDEs for The Move     Save   ║
║                                                              ║
║  Extension:    HTML Element:   User-Defined Extensions:    Save As ║
║                                                              ║
║  ┌──────┐    ┌──────────▼┐  │Element│ HTML │ Description │        ║
║  └──────┘    └───────────┘  │Committe...│ DIV │ Member of the mov... │ Help ║
║  Description:                │SpecialA...│ EM  │ Special Attention n... │    ║
║                                                              ║
║  ┌────────────────────────┐                                  ║
║  └────────────────────────┘                                  ║
║       ┌─────────────┐                                        ║
║       │   Add >>     │                                        ║
║       └─────────────┘                                        ║
║       ┌─────────────┐                                Exit    ║
║       │ << Delete/Edit│                                       ║
║       └─────────────┘                                        ║
╠══════════════════════════════════════════════════════════════╣
║ C:\Program Files\SoftQuad\HiP\Help\tutorial\samples\helpers\the-move.hpe ║
╚══════════════════════════════════════════════════════════════╝
```
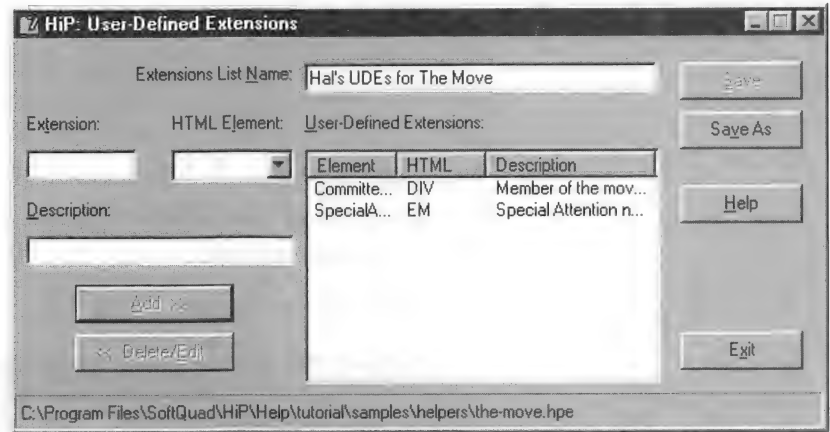
There are at least three things you will need to specify for a UDE. The first is the title of the file in which your UDEs will be stored—the **Extensions List Name**. This title will appear in the **Document Properties** dialog. A typical use for it would be to identify a logical grouping of UDEs. (This is the title of the file, not the file name.)

The next thing you will want to do is decide on a name for the UDE itself. This is the name that will show up as a tag icon in the H.i.P. Editor when you are editing a document and choose to apply this UDE. Let's say you decided on the name 'DuckSoup'. You would then see **DuckSoup** as one of several elements available when you choose **Insert Element...** from your **Markup** menu in the H.i.P. Editor. Since the UDE DuckSoup will be listed among the many standard HTML elements, you may want to follow a convention when you are naming them—just so that you can distinguish UDEs from the other HTML elements. Using mixed-case UDE names might be the best choice. It would appear this way on your editing screen: (DuckSoup⟩ .

The next choice you will need to make is the HTML element on which your UDE will be based. The choices, presented in the HTML drop down list, are the elements that are common within the **BODY** element of your documents. However, you can choose any HTML element by typing its name in the HTML element text box. Your choice should be based on the

similarity in use of your UDE to a common HTML element. You'll notice that Hal chose EM as his HTML base for SpecialAttention.

Once you have an Extension name and an HTML base element, you will want to describe its use. This Description shows up in the H.i.P. Editor's Insert Element dialog.

Hal has already created the SpecialAttention UDE for us, but we can add a new one of our own, just to practice using the User-Defined Extensions dialog. Hal likes ducks so let's use 'DuckSoup':

- Type DuckSoup in the Extension field .
- Choose DIV from the HTML Element list box.
- Type Favorite Movie in the Description field.

This is how your dialog box will look.
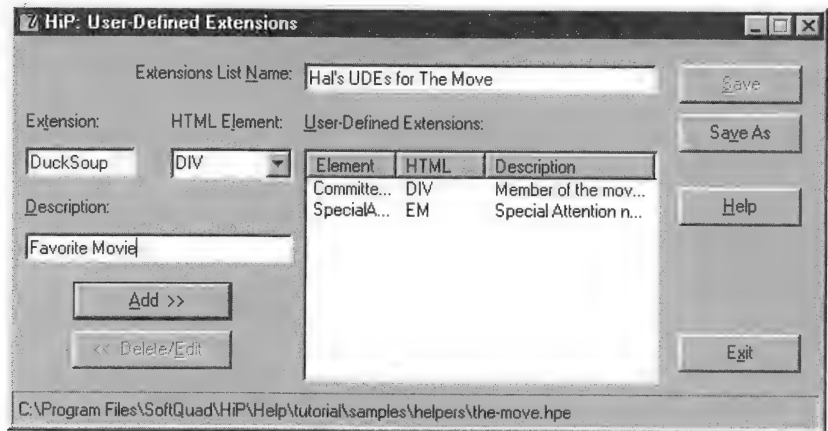
- Click on the [Add >>] button
- Click on the [Save] and [Exit] buttons.

As a UDE, DuckSoup might have a fairly limited use in the metal-forming business. But, if you were creating a page of Groucho Marx quotations, you might use this kind of UDE to identify which Marx Brothers movie contains a particular quote. In your H.i.P. Editor, such an entry might appear this way: ⟨DuckSoup⟩ A child of five would understand this. Someone go fetch me a child of five. ⟨/DuckSoup⟩
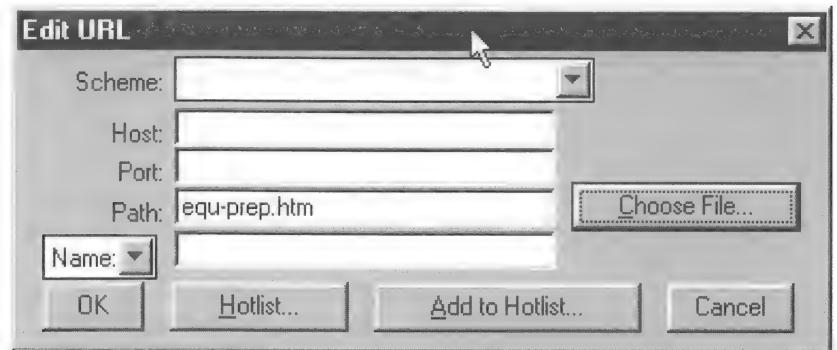
Hal knew of several instances where he would use his SpecialAttention UDE himself, but he also wanted to make sure that all of the authors who would contribute documents to the project were encouraged to use it too. If it were used consistently, Hal knew, all of the project's readers would automatically be able to compile lists of moving problems and special tasks.

To demonstrate the usefulness of his new UDE, Hal took a document that he had been sent by the manager of the forging plant. It was a set of instructions and cautions for preparing and shipping of the plant machinery. A sample of the document is available for you to import into the project:

- Switch to the H.i.P. Information Manager.
- Choose **Import File...** from the **File** menu.
- Click on the [Browse...] button.
- Navigate to the *samples* folder in the *tutorial* folder, and select *equ-prep.htm*.
- Click on [Open] .
- Make sure that the **Destination** is *the-move*.
- Click on [OK] .

Once the file *equ-prep.htm* is imported, you may want to create a link to it:

- Open *index.htm* (also known as 'The Move Home Page') in the H.i.P. Editor, (right-click on it and choose **Edit File.**
- Insert a new **P** element and place the insertion point inside it.
- Click on the toolbar button.
- Click on [Choose File...] button.
- Navigate to *the-move* folder and select *equ-prep.htm*.
- Click on [Open] .
- Click on [OK] in the **Edit URL** dialog.

```
Edit URL                                                    ⊠
      Scheme: [                              ▼]
        Host: [                                ]
        Port: [                                ]
        Path: [equ-prep.htm                   ]   [ Choose File... ]
    [Name: ▼] [                                ]
    [  OK  ]  [ Hotlist... ]  [ Add to Hotlist... ]  [ Cancel ]
```

- ☐ Type `Equipment Preparation Guidelines` within the A element.
- ☐ Save the file.

There's no need to remap the cyberbolic display—the Information Manager and the H.i.P. Editor have already talked to each other. Now you can open *equ-prep.htm* in the H.i.P. Editor and do some markup using the new SpecialAttention UDE.

Hal has already done some editing on the document and has marked up a a few of these special problems, but you may want do a couple more for practice.

This is how you might proceed. First, link the document to Hal's UDEs:

- ☐ Open *equ-prep.htm* in the H.i.P. Editor..
- ☐ Click on the [ ] toolbar button.
- ☐ Click on the UDEs tab.
- ☐ Click on the [Browse...] button.
- ☐ Navigate to the *helpers* folder in *the-move* folder.
- ☐ Select 'Hal's UDES for The Move'.
- ☐ Click on [OK] in the Browse for UDE file dialog.
- ☐ Click on [OK] in the H.i.P. Document Properties dialog.

Next, we'll find the items that need to be marked, and surround them with SpecialAttention.

We will mark up:

- □ limit switch test will be carried out
- □ preparation of the electrical transformers
- □ training session on oil and solvent handling safety
- □ area of the plant will be designated for the temporary storage of oils and solvents

- ■ Click on the find button 🔍 on the toolbar.
- ■ Find and select the phrase that will be surrounded by the SpecialAttention UDE.
- ■ Press `Ctrl-I` to insert the markup.
- ■ Select **SpecialAttention** (you may have to scroll down the list).
- ■ Repeat this procedure for all four phrases (above). The last one you mark will look something like this:
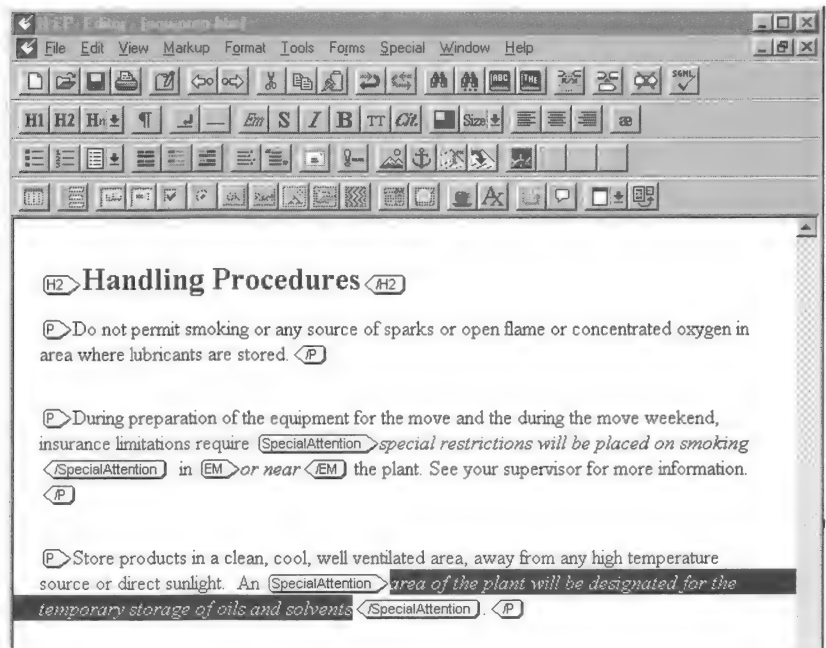
■ Save the file and click on one of the browser buttons on the toolbar.

When you are finished marking up the document with Hal's 'SpecialAttention' UDE it will look like this in your H.i.P. Viewer:
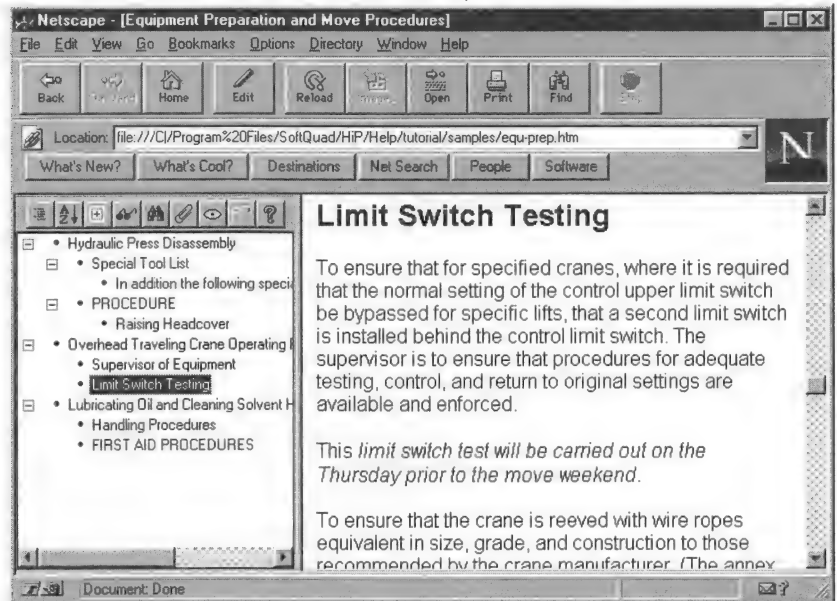
You were expecting it to look different weren't you? So was Hal. In fact, when it didn't blink or pulse or glow, he was a little disappointed. Of course, the H.i.P. Viewer *did* show the text in italic face, but that was really no different from the text marked with EM for *emphasis*. "So," thought Hal, "what's the big deal?"

But there's more to this than meets the eye. The inner beauty of a UDE is that, once it is defined and applied, it can be used to effect processing decisions of almost any kind. Whether you want them to color themselves red, or appear as items in an index, or print in a certain way, they will always be there to designate a 'particle' of your information systems's atomic structure.

For now, that may just mean appearing in a Live TOC, but in Exemplar's future automated text database systems will almost certainly be used, and well-structured text will make that path much easier to follow. (Hal was a little skeptical about all that—he just wanted to get this stuff on its way to Neoteric.)

By the way, the reason this new UDE does not show up automatically in the default Live TOC is that it is behaving like the EM on which it is based. Having the content you have marked with your UDEs show up automatically in the default Live TOC is something you might consider to be pretty handy. On the other hand, you might not. Making Live TOCs work the way you want them to is the subject of the next section.

Creating a Live TOC

Hal decided that a Live TOC would be just the item to show off his new 'SpecialAttention' UDE. Of course, he had already gained some experience with Live TOCs; a Live TOC was created automatically, for example, when Hal converted Sally Wong's sales office moving instructions. H.i.P. takes the common headings (H1, H2, H3, etc.) and makes a multi-level table of contents. That's how the TOC/index for the moving instructions was built before Hal even knew about Live TOCs (see page 56).

But this Live TOC was going to be different. Every time anyone viewed a document on the moving site and chose this custom Live TOC, they would be treated to a 'special attention' list generated on-the-fly. It would appear in the Live TOC panel of everybody's H.i.P. Viewer. Every time a new problem item was discovered, the document's author could tag it for everyone else to see—and then, when the problem was solved, untag it.

Since he had based 'SpecialAttention' on an EM element, it was not automatically picked up in the default Live TOC—the trick would be to set up another Live TOC that would display the list of potential moving problems.

Hal's Live TOC 'extracts' the SpecialAttention UDE and ignores everything else. Yours will do the same. There are two processes to complete: create a new Live TOC in the Live TOC editor, and link it to the document in the H.i.P. Editor—make that three steps: you'll want to view your work in the H.i.P. Viewer.

First Hal created a new Live TOC:

- Switch to the H.i.P. Information Manager.
- Choose **Live Tables of Contents...** from the **Tools** menu.
- Navigate to the *helpers* folder under *the-move* folder.
- Type `specattn` in the **File name** text box of the **Open Live TOC** dialog, (let H.i.P. take care of the file extension).

Since it's a new file name, you will be asked if you want to create the file.

- Click on [Yes] .
- Type `Special Attention` in the **Live TOC Name** text box.

Since the element that we want to use in our Live TOC is a UDE, the Live TOC editor doesn't know about it automatically—it has to be loaded explicitly:

- Click on the [Load UDE...] button.
- Select the file *the-move.hpe* (that's the file that contains the UDE) from the *helpers* folder in *the-move* folder, and click [Open] .
- Expand the drop down **Elements** list on the **Live Table of Contents** dialog , and select **SpecialAttention**. (It's way down at the end of the list—you'll have to scroll.) Note that **CommitteeMember** is also in the list.
- Click on the [Add >>] button. This copies SpecialAttention to the list of Live TOC elements.

Live Table of Contents

LiveTOC Name:

Special Attention

LiveTOC Elements:

SpecialAttention

Elements:

cialAttention ▼   Add >>

Delete

Save

Save As...

Load UDE...

Help

Exit

C:\Program Files\SoftQuad\HiP\tutorial\the-move\helpers\specattn.hpl
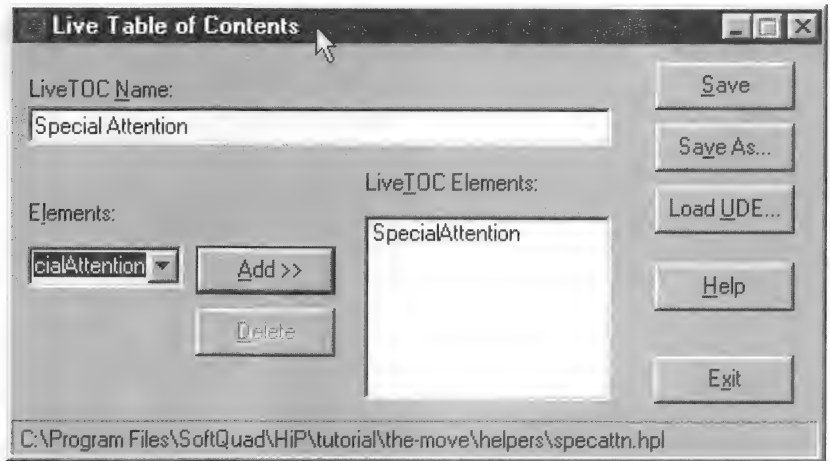
- Click on the ⌈ Save ⌋ button, then the ⌈ Exit ⌋ button.

Now we have a new Live TOC in its own TOC file called *specattn.hpl*. The next step is to link it to the document that will have the SpecialAttention UDEs. This is very similar to what you did when you linked the document to the UDE file (page 75).

You probably have *equ-prep.htm* open in the H.i.P. Editor, so we will do the next operation there. But, you could also edit properties from the H.i.P. Information Manager.

- Switch to the H.i.P. Editor. You may still have the file open, but, if not, open *equ-prep.htm* from *the-move* folder.
- Choose Properties... in the File menu.
- Click on the Properties tab, then on Live TOCs in the tree diagram.
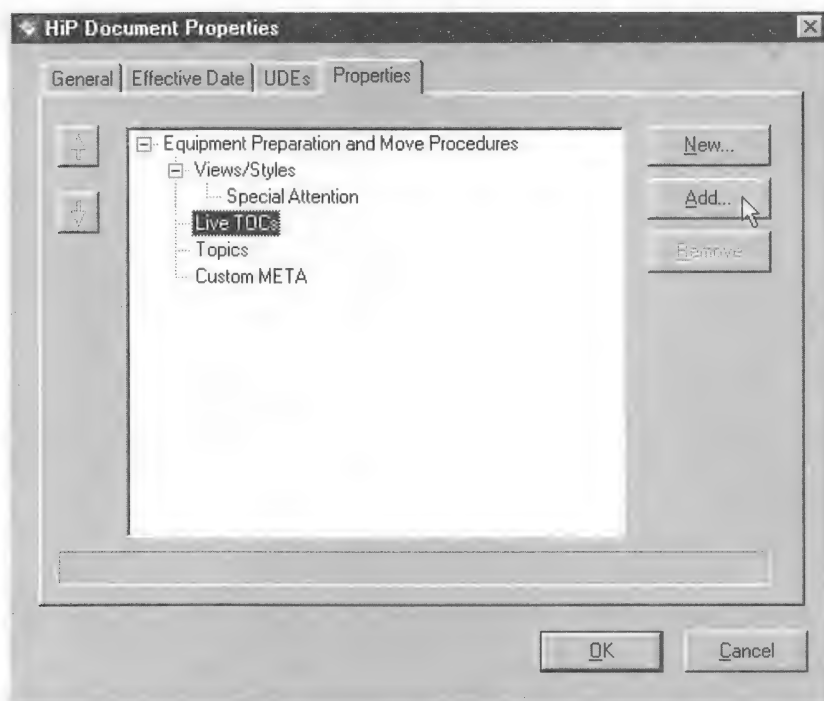
- Click on the [Add...] button to link to a Live TOC file.
- Navigate to the *helpers* folder in *the-move* folder.
- Click on the check box beside 'Special Attention'. Note that you are selecting the title of the file, not the file name.
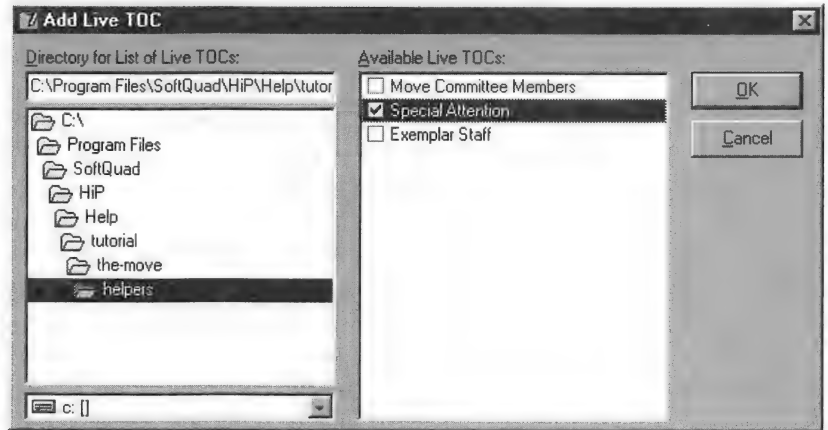
- Click on the ⟨ OK ⟩ button in the **Add Live TOC** dialog, then again in the **Document Properties** dialog.
- Save your file.

Now, the third step: view your document. When *equ-prep.htm* appears in the H.*i*.P. Viewer, the TOC display will show the 'Special Attention' contents. Since this is a 'custom' Live TOC it takes precedence over the default Live TOC—though the default **Full View** is still available by clicking on the **Choose Live TOC** button ⊞ . Choose **Special Attention** again.

Now the TOC panel lists only the items needing special attention. Clicking on one will take you directly to the part of the document that describes the problem. Try a few.

Before you go on to the next section, you may want to close your document and minimize the H.*i*.P. Editor.

Distinguishing one view from another

Hal didn't like writing any more than most of us, and he liked checking somebody else's work even less. The idea of taking one piece of text and making it do a job-and-a-half had a strong appeal for Hal, whose motto was, "Laziness hatches invention." Exemplar's Director of Human Resources, Herb Klamm, was the opposite—he was the kind of guy who preferred three words where one would do, and felt there was hardly any problem that couldn't be solved with a policy memo. When Herb Klamm got wind of the almost infinite capacity of the intranet you had to notice the new spring in his step.

Within hours, Herb was producing text files that re-refined the description of Exemplar's relationship with its employees. When he delivered the files to Hal for publication he was surprised with the response: Hal was delighted. This was exactly the kind of information that people would find any excuse not to read. And Hal was going to give them one.

Despite Herb's best efforts, employees would be pleased to ignore most of the material most of the time. "So," asked Hal, "why not reduce, reuse and recycle? Instead of..." he thought to himself, "replicate, recreate and reiterate."

Hal showed Herb how the H.i.P. Editor could be used to tag sections of his files so that his 'Procedures and Policies' (or was it 'Policies and Procedures'?) document could be broken down into 'Policies' and 'Procedures'. With two Views, Herb Klamm's documents would get straight to the point. "This," thought Hal, "is truly a benefit to humankind." He sent Herb off to systems to get SoftQuad H.i.P. installed in Human Resources.

Views do not prevent readers from seeing content they shouldn't be reading—they are not security devices. But they do something that might be just as important: they prevent readers from having to dig through information that is meaningful only to someone else.

In the meantime Hal marked up a sample of Herb's document. We won't bore you with a before-and-after, but if you'd like to see how the reader sees (or doesn't see) views:

- Import *handbook.htm* into your project (the *samples* folder in the *tutorial* folder). You will get an **Existing File** warning regarding *mail.gif*.

- Click on [Cancel] , allowing the import to proceed. (It doesn't matter if you accidentally overwrite *mail.gif* or create a file with a new name.)
- Click on [Yes] when you are asked to confirm the cancellation.

Once the document is imported, we can create some useful links:

- Select *handbook.htm* in the Project display.
- Click on the [icon] button.
- Click on the **Properties** tab and then on **View/Styles**.
- Click on the [Add...] .
- Navigate to the *helpers* folder in the *tutorial* folder.
- Click a check mark into the check boxes beside the 'Policies and Procedures' styles.
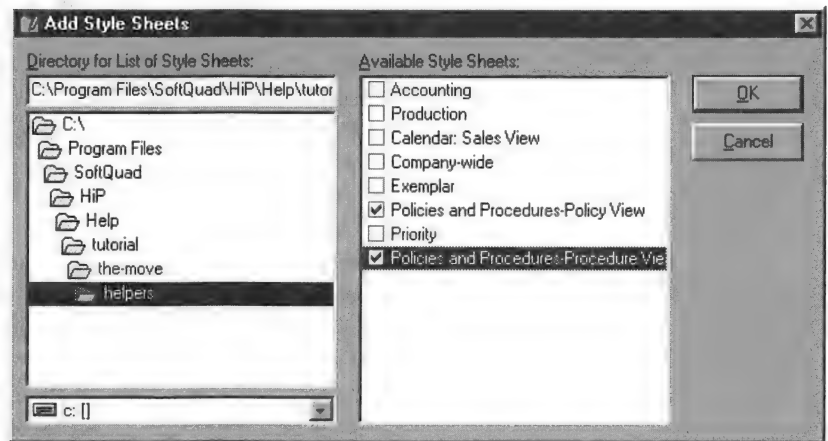
- Click on the [OK] button in both the **Add Style Sheets** dialog and in the **Document Properties** dialog.

Now Herb's document is ready to view:

- Right-click on *handbook.htm* in the Project display and choose **View in Browser**.

Once the document is loaded into your H.i.P. Viewer:

- Click on the Views button in your H.*i*.P. Viewer, .
- Choose **Procedures** from the menu.

In this case, Hal chose to create views using class values. He could have made a couple of new UDEs instead, and we'll look at that method too in the next section. In the meantime, let's mess up Herb's document by designating some text randomly with class values of either 'Policy' or 'Procedure'.

- Switch to the H.*i*.P. Editor.
- Choose an element that you can use to surround blocks of text—a DIV is a typical choice.
- Select a piece of Herb's Policy text and surround it with a DIV.
- Type F6 or choose **Element Attributes...** from the **Markup** menu.
- Type Policy in the **CLASS** text box.

Here you are creating a *class*, which is similar to a UDE, but more general. Elements in a document that share the same CLASS attribute are all members of the same class. This is the case even if some of the elements are DIV and others are P or any other element type. Unlike a UDE, a class doesn't have to be explicitly defined—you can just choose the element and give the correct class attribute. The primary use of classes in SoftQuad H.*i*.P. is to create document view.

For example, you may have entered 'Policy' as the CLASS of several Table elements: now you discover that there are some headings that should also belong to the 'Policy' view. Implementing this is simple: give their CLASS attribute the value, 'Policy' too. Just press F6 , and type it in. Distinguishing one type of text from another using classes (instead of the UDEs) makes sense when the text you need to distinguish is found in a number of different elements—headings, paragraphs, tables, etc. You can change the CLASS attribute of whatever element happens to contain the text you need.

Designating elements as either 'Policy' or 'Procedure' is half the process. The next step is to create a style that treats these element-class types in different ways. (Of course, some of the text may be neither.)

A first look at cascading style sheets

Before you get started, you should understand at least as much as Hal did about cascading style sheets. At first, Hal found the whole CSS thing a bit baffling—and so it might seem to you. In fact, it's a completely new idea for intranet and internet viewers and you are right here at the leading edge. So, the first time through, let's just take what we need. For a detailed description of the CSS Editor, see the *Styles and Views* chapter of the *Reference Guide*.

If you've had any experience with style sheets in word-processors, you know that they're pretty simple conceptually: someone decides how everything in the document will look and gives each look (or 'style') a name. For example, all of the addresses in a document might be presented in 10 point type, single line spacing, in a Lucida Handwriting type face (OK, fine, Arial then). So we'll call that style 'address' and whenever somebody types in an address, they will choose that look from the style sheet and all of the addresses will look similar.

The same idea is being used for documents that are delivered to Web browsers and H.i.P. Viewers. When the document is delivered to the browser, a stylesheet is delivered with it. The rules on the stylesheet control the way things look in the H.i.P. Viewer. But, the rules don't always have complete authority. Sometimes, the stylesheets allow particular rules to be ignored in favor of other choices when the document reaches a particular browser. So the rule might be stated like as follows: 'this is an address, addresses must be presented in 10 point, single-spaced type, and that type should be Arial—but, other typefaces can be chosen'. Somewhere down the line, this rule is received by a browser that says 'Yes! At last an opportunity to use my all-time favorite, Lucida Handwriting!' This permission to override the style sheet leads to the name 'cascading style sheets'—because the line of authority 'cascades' down to the browser.

"But what," Hal demanded, "does all this have to do with making Views for documents?"

It's true that document styles are usually thought of as rules about typeface, size, indentation, maybe color. Well, that last one, 'color', gets us close. What if the color were 'invisible'? Then you would have content that could not be seen. That's the trick. A View is merely the application of a style rule that says, for example, 'everything in this element should be made invisible.'

Now, let's try making a stylesheet to hide some of the information in a document—information that is, for a particular set of readers, just unnecessary clutter. Let's use that document that Sally Wong sent from the sales office. In her case there are two sales offices that will be moved to the same location, but by different moving companies. Each company has its own policies, labeling methods and packing box types. You may want to view the document (*slspack.htm*) to refresh your memory. When you're ready to begin creating a new View of the document:

- Double-click on *slspack.htm* in your Project display, or highlight it and click on it with your right mouse button, then choose **Edit File.**

In the section above, 'Distinguishing one view from another' (page 88), Hal chose to create two views of Herb Klamm's HR document using CLASS attribute values. He assigned different attributes ('Policy' and 'Procedure') to elements that contained the text he wanted to control—either hiding it, or revealing it. In this example, we're going to use a different method—we're going to create two new UDEs instead, then set up the style sheet to act on them. There are good reasons for each method. In this case, where our text is mainly in paragraphs, we will need only two UDEs. If our text were contained in many different elements, it would be tedious to create UDEs for each one.

So, the idea here is to make a UDE for the information that is intended for the exclusive use of Exemplar's people and a second one for information that is exclusively aimed at the Priority people. Everything else—all of the non-exclusive information in the document—is left outside of these two UDEs.

The first time you do this, it will seem complicated, but once it's set up, you will find that documents that have overlapping information can be maintained much more easily because you won't need multiple, altered copies for each different audience.

Let's separate the Exemplar sales staff from the Priority sales staff and make sure they each get their appropriate instructions. The first part of the process is to enclose both the Priority-specific and the Exemplar-specific text with UDEs:

- Switch to the H.i.P. Information Manager.
- Choose **User-Defined Extensions...** from the **Tools** menu.

- Select the file *the-move.hpe* from the *helpers* folder under your *the-move* folder and click on the ⟨ Open ⟩ button.
- Make a new UDE called 'HiddenFromPriority' based on the DIV element. (If you need to review UDE creation, see page 75.)
- Make a second UDE called 'HiddenFromExemplar', also based on DIV.
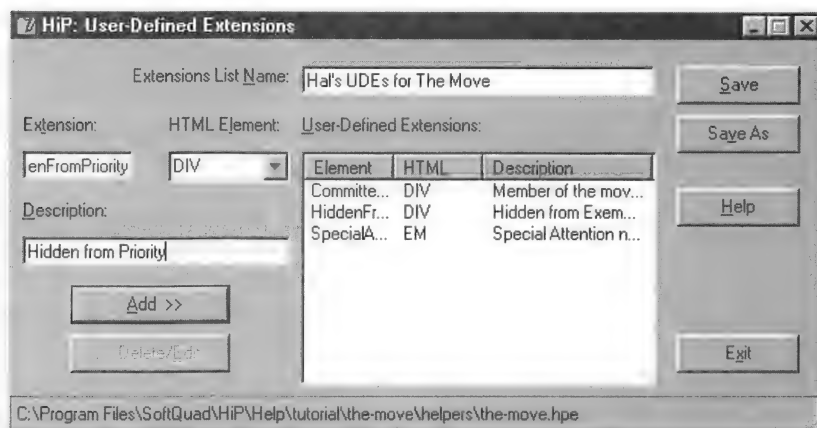
- Add it to the list.
- Save your *the-move.hpe* file and close the dialog.
- Close the dialog by clicking on ⟨ Exit ⟩ .

Now that you have two new UDEs, you can edit Sally Wong's document and apply each UDE to the appropriate content. The document itself will need to have its properties set to link to *the-move.hpe*.

- Switch to the H.i.P. Information Manager and right-click on *slspack.htm*.
- Choose Edit File from the menu.

Once the file has been opened in the H.i.P. Editor:

- Chose Properties... from the File menu.
- Click on the General tab and make sure the file has a title.
- Click on the UDEs tab and then on the ⟨ Browse... ⟩ button.
- Navigate to the *helpers* folder under your *the-move* folder.

- Select 'Hal's UDEs for The Move'.
- Click on the ⌐OK⌐ button.
- Click on the ⌐OK⌐ button in the **Document Properties** dialog.

The next step is to select the text to surround in each of the two new UDEs that you have just made available to the document.

- Find the paragraph that begins 'Staff at Priority' and surround it with a HiddenFromExemplar UDE.
- Find the paragraph that begins 'Exemplar people' and surround it with a HiddenFromPriority element.
- Continue through the file surrounding text as you see fit.

We have divided the content into three categories—Priority, Exemplar, and both (i.e., not marked up with either of the two 'Hidden' UDEs). Now we can use different Views to see the same document in different (but not completely different) ways. Fortunately, Hal has been here before us, so there are already a couple of Views on hand—all we have to do is select them:

- Choose **Properties...** again from the **File** menu.
- Click on the **Properties** tab and then click on **Views/Styles** in the tree diagram.
- Click on the ⌐Add...⌐ button and select the *helpers* folder under your *the-move* folder.
- Click on the check boxes for 'Exemplar', and 'Priority'.
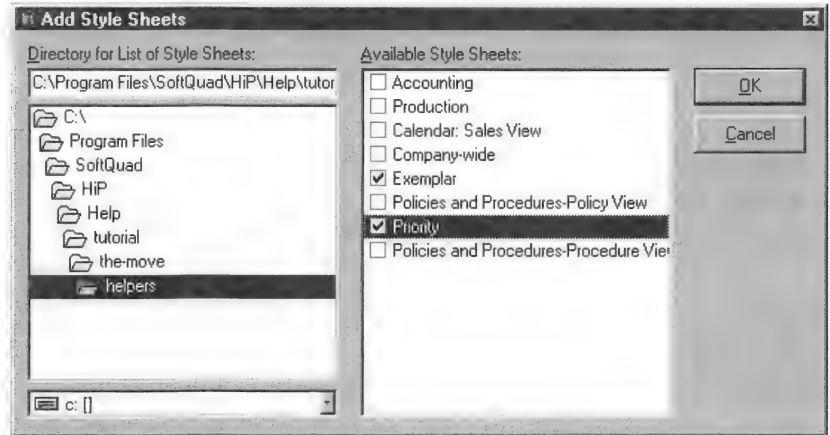
- Click on the [OK] button in this dialog, and in the **Document Proper-ties** dialog.

Now the two cascading style sheets that contain the instructions to hide Exemplar and Priority information are linked to your document.

Ready to see the results?

- Save the *slspack.htm* document that you have been editing.
- Switch to the H.*i*.P. Information Manager.
- Click on *slspack.htm* in the Project display.
- Click on the [Browser] Browser button.

When the H.*i*.P. Viewer has loaded the document:

- Click on the [👓] button at the top of the Live TOC panel.
- Choose **Priority** from the menu.
- Check to see that the packing instructions are the ones intended for the Priority sales office.
- Choose **Exemplar** and check the instructions again.

Of course, any of the sales people who really wanted to read how the other sales office is packing their moving boxes could do so. There's no security implied by views; it's more like dividing a newspaper into a

sports section and an arts and entertainment section—it appeals to an interest without necessarily restricting information.

Topics are easier to find

Now that some of Hal's pages were getting larger, he had a clearer sense of what it might be like for a new reader to find her way around. Even though he was familiar with much of the content, Hal noticed that he was occasionally unable to find things he was sure should be there. It was baffling. For example, he had been given a document that he knew included some text announcing that pre-printed labels for packing boxes would be available at the reception areas of each of the company's locations. He had to find that section and place it on an announcement page. His searches on 'label' and 'moving label' turned up nothing. But then he stumbled on the document and realized that it talked about 'tags'.

It's a familiar problem to anyone who has had to develop an index for a book. And there's a familiar solution: make sure that a search on a plausible synonym directs the searcher to the correct location. H.i.P. Topics were Hal's answer because they allow the addition of keywords to a document.

Since topics are 'added' to a document instead of being 'selected' from it, there is an opportunity to create another set of hooks to allow people to find things. Some of Hal's readers would call them 'tags', others 'labels', and some might say 'stickers'. In the end, he added 'tickets' just in case, even though that word didn't occur in any of the files. Now, anyone can find the information on where to get their packing tags/labels/stickers/tickets and how to fill them out—even if they search on the 'wrong' synonym.

Adding topics to a document is a two-step process—at least the first time. Step one is to create a file that contains the topics that will be useful for a project. The second step is to link a document to the topics file and choose the appropriate topics. You can do both steps in either the H.i.P. Editor or the H.i.P. Information Manager.

In this case, we'll use the H.*i*.P. Information Manager :

- Switch to the H.*i*.P. Information Manager.
- Import the file *tagmemo.htm* from the *samples* folder. (Notice that the cyberbolic display is indicating a broken link to *tagmemo.htm*—there's a link to it within the *index.htm*.)
- Highlight the file name again in the Project display.
- Click on the ▨ button.
- Click on the **Properties** tab in the **Document Properties** dialog.
- Click on **Topics** in the tree diagram. (Notice that the New... and Add... buttons become active.)
- Click on New... .

When the **Edit Topics File** dialog appears, you will need to write a title, then write and add your topics. When you are finished, you will have a new project helper file (with an *.hpo* file name extension). You can also link other documents to this topic file. To complete your new topic file:

- Type a title into the **Title of Topics File** text box. 'The Move' would do because this file will contain topics for the whole project.
- Type the topic `tag` into the **Topics** text box and click on the Add>> button.
- Repeat the process for 'label' and 'sticker'.

This makes the three topics that you will add to the *tagmem.htm* file. But other files will also use this topic file, so throw in a few more topics while we're here.

- Repeat the add process for 'move date', 'conference', 'solvents', and 'office plants'.
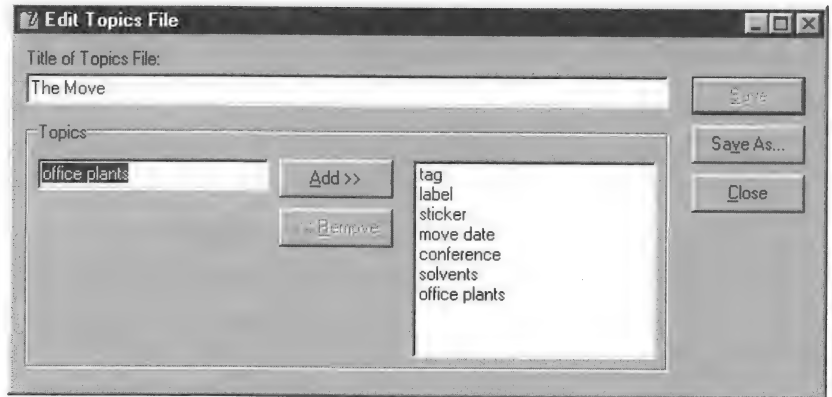
- Click on [Save As...] and save the file as *the-move.hpo* in the *helpers* folder in the *the-move* folder.
- Click on the [Close] button.

This brings you back to the H.i.P. Document Properties dialog. Note that there's nothing new in the tree diagram: we haven't yet added our new topics file yet:

- Make sure that Topics is highlighted and click on the [Add...] button.

The Add Topic dialog appears. Look at the bottom of the dialog where you will see Find list of Available Topics in. There are two buttons, [Browse...] and [New...] . We already have a topic file, so we'll just browse to find it:

- Click on the [Browse...] button.
- Navigate to the *helpers* folder under your *the-move* folder.
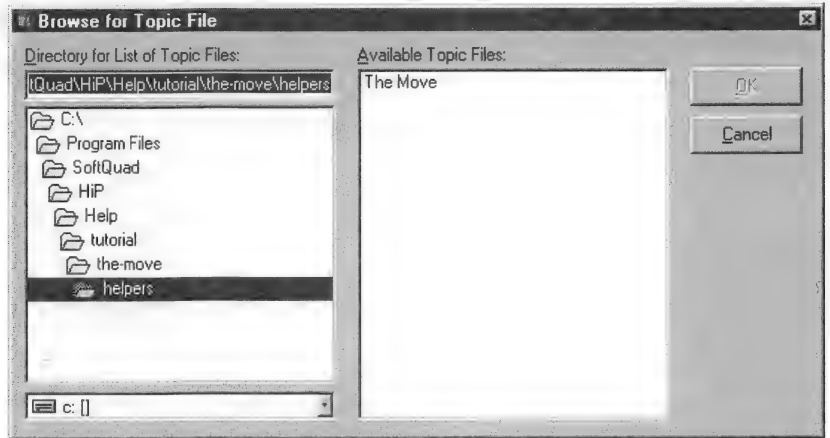
- Select **The Move** in the **Available Topic Files** list.

Notice that we are choosing the file by the title we gave it, not by its file name. Most authors of your Web site documents will see helper resources in this way—by title, rather than by file name.

- Click the [OK] button.

This choice of topic file makes available to your document all of the topics that you wrote—including some that you don't want for this particular document. Choose the ones you need:

- Do a multiple select. If the topics you need are next to each other hold down your [Shift] key while you click on each topic. If they are separate, hold down your [Ctrl] key while you click.
- Click the [Add>>] button to create a list in the **Selected topics** box on the right of the dialog.

At this point, you might have second thoughts: what if someone calls this thing a 'ticket'? Better add a third synonym.
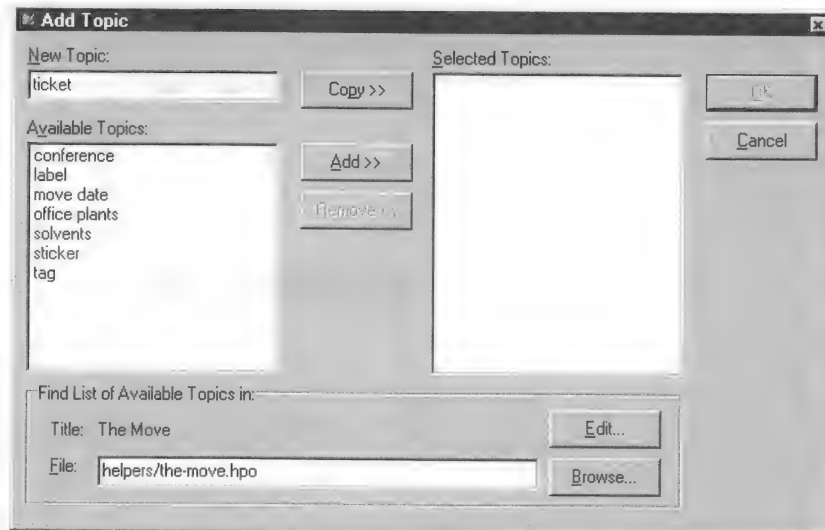
- Type ticket into the **New topic** text box and click on the [Copy>>] button.
- Click on the [OK] button in the **Document Properties** dialog.

Now the tree structure of the **Document Properties** dialog shows some new information.

- Click on the [OK] button.

That was a lot to remember, so let's stop and summarize. Here's what we did:

☐ Used the H.i.P. Information Manager (we could have used the H.i.P. Editor) to pick a document (*tagmemo.htm*) that needed some topics.

☐ Created a separate (but linked) file to contain topics, not just for the *tagmemo.htm*, but for other documents in the project.

☐ Created several topics—some that we needed for *tagmemo.htm* and some that would be useful elsewhere in the project.

☐ Saved the new topic helper file (*the-move.hpo*) in the *helpers* folder.

☐ Linked the 'The Move' (that's the title we gave to *the-move.hpo*) to our *tagmemo.htm* document.

- ☐ Chose three topics from the topics file and added them to the document.
- ☐ Thought of a fourth topic and added it on the spot.
- ☐ Saved the file *tagmemo.htm* with its new topics.

Now whenever a user searches on a topic, or asks the H.i.P. Monitor to send notification of a change to a document that refers to a particular topic, the information will be forthcoming.

By the way, the *tagmemo.htm* was put together in a snap. Hal found a template for an announcement. To find one for yourself, in the H.i.P. Information Manager:

- ■ Choose **New File...** in the **File** menu.
- ■ Choose **core** from the **Template Group** list box in the **Choose Template** dialog.
- ■ Select *announce.htm*.
- ■ Click on [OK] .

So go ahead, announce something!

Making things pop up

By this time, Aleksandr was back with a series of photo gallery files. He had scanned the originals as GIFs, and had placed the photos of people in each work group together in one file. There were a couple of dozen files. Apparently Hal was not the only Exemplar employee who had learned to be H.i.P. .

Aleksandr also did something Hal had not expected—he had added text information about all of the people in the photos, but he had not done it in the obvious way, simply writing a few lines under each photo. Instead, he had created a pop-up to contain each text description. The effect made the photo gallery a completely different kind of information source for its readers. Instead of being an earnest, serviceable presentation of faces and names, it had become a game. In fact, Aleksandr had titled the starting page 'Test Your Who's-Who Know-How', and had invited readers to guess the name, responsibility, and workplace before clicking the pop-up button.

- Switch to the H.i.P. Information Manager.
- Import *photgall.htm* from the *samples* folder in the *tutorial* folder.
- View the file *photgall.htm* (right-click on it and choose **View in Browser** from the menu).
- Click on the line 'Sales Group'.

Pop-ups seemed to open up several possibilities for other documents in the project. Packing information into a pop-up allows documents to be presented with less clutter and helps readers keep a sense of their place in the document—that's especially important when there are many links to follow and it's easy to get lost in the details.
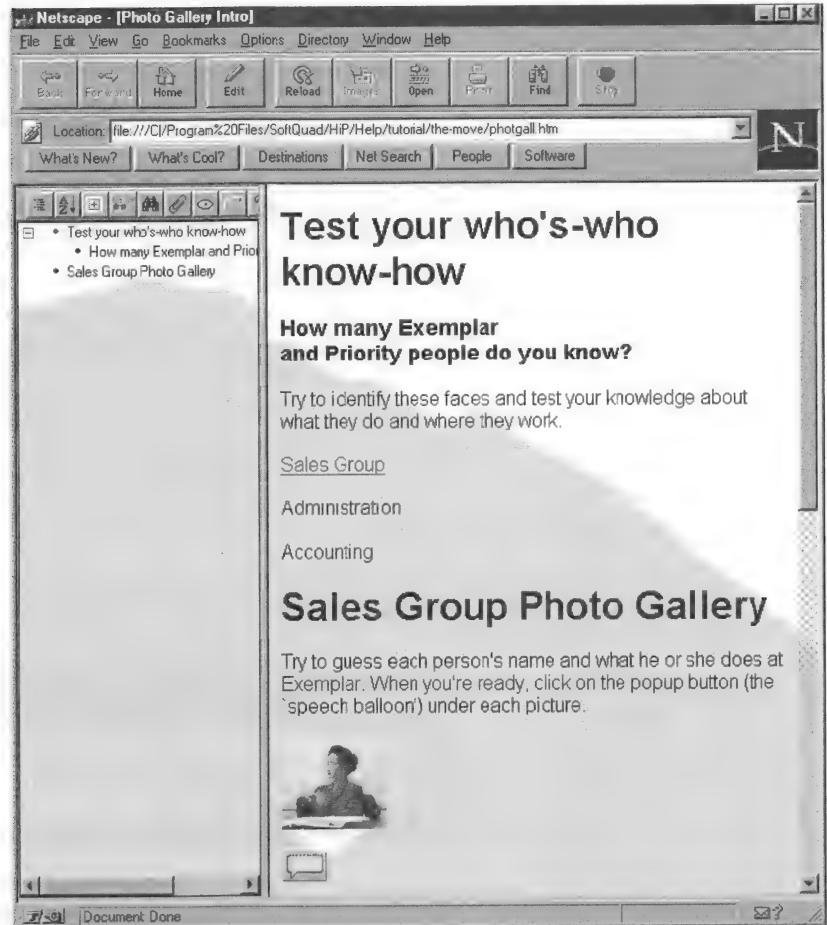
Hal asked Aleksandr to show him how it was done and was surprised at how simple it was. They opened a 'frequently-asked-questions' document (a FAQ) that contained questions and answers. Hal thought the FAQ might be easier to use if the answers appeared in pop-ups.

The first couple are done; you might add the others:

■ Import the file *move-faq.htm* from the *samples* folder in the *tutorial* folder.

- Open the file in the H.*i*.P. Editor and find an answer that hasn't yet been surrounded by a BlockPopup element.
- Click on the opening P tag to select the answer.
- Click on the ▯ pop-up button on the toolbar.
- Surround the rest of the answers with BlockPopup elements.

When Hal previewed his pop-ups, his FAQ document contained questions followed by pop-up buttons. When you view the document, click on the pop-up button and see what happens.

This particular pop-up is a block pop-up—it is displayed in its own window. In the document, a BlockPopup usually surrounds a 'block' of text rather than just a few words in a line. After he had created a couple of these 'block pop-ups', Hal realized that, since they were just containers for H.*i*.P. contents, they could contain many of the elements that you might put in a H.*i*.P. document—including links. That's when he added the link to the calendar and the e-mail address for the softball team organizer.

There's a second variety of pop-up—the 'inline' pop-up (there is one in the Photo Gallery page for the Sale Group—at the bottom).

The pop-up button is similar to the one for block pop-ups when viewed in the H.*i*.P. Viewer, but, the pop-up itself is a banner of text just below the position of the button. Inline pop-ups are useful when the explanation can be very brief and need not lead the reader off in another direction. Try making one:

- Switch to the H.*i*.P. Editor with *move-faq.htm* open.
- Place the cursor just *after* the phrase 'softball diamond'. (By the way, you can't place an InlinePopup inside a BlockPopup.)
- Click on the ▯ button on the toolbar.

- Write your own definition of 'softball diamond' (try to limit it to 10 words) in the InlinePopup element.
- Save the file and preview it.

Note the button ⊡ . When you click on it, you will see your softball diamond definition.

At this point, you many want to switch back the H.i.P. Editor and save, then close your file.

Pop-ups make it much easier to keep the clutter out of your text. People who know, or don't need to know, can keep their train of thought on track while reading over a pop-up button. Everyone else gets a deeper explanation when they feel they need it. It's just a better way of organizing information—and organizing is the subject of our next chapter.

Getting organized

With several new documents in place in his 'The Move' project, Hal was getting to the point where he couldn't keep track of it all in his head. Of course, the H.i.P. Information Manager made it easy not to lose documents and it also revealed linking problems with its cyberbolic display. But Hal was getting the sense that some of this information would have a life beyond the company's move and should, perhaps, be organized in such a way that it could be separated and reused after 'The Move' site was no longer needed.

Maintaining links despite copying and moving

The thought of starting a new project didn't worry Hal, but he knew that the information was still developing and he didn't want to have to maintain completely separate projects. The solution, he found, was to place certain groups of files in sub-projects—really just subdirectories on his disk.

The photo gallery is one group that would have a useful life long after the move, so Hal used the H.i.P. Information Manager to collect all the photo pages and move them to a new sub-project that he named *gallery*. First, we'll create a new folder called *gallery* in *the-move* folder:

- Select the first line in the Project display, *the-move*, (it has an ▪icon beside it) and choose **New Project** from the **File** menu.

- Click on the **New Project** radio button and on the ⌈ Next>> ⌉ at the bottom of the Wizard dialog.

- Select *the-move* as the folder in which you will place your new folder.

- In the text box, place your insertion point at the end of the path name and type: `\gallery`.

- Click on the ⌈ Finished ⌉ button.
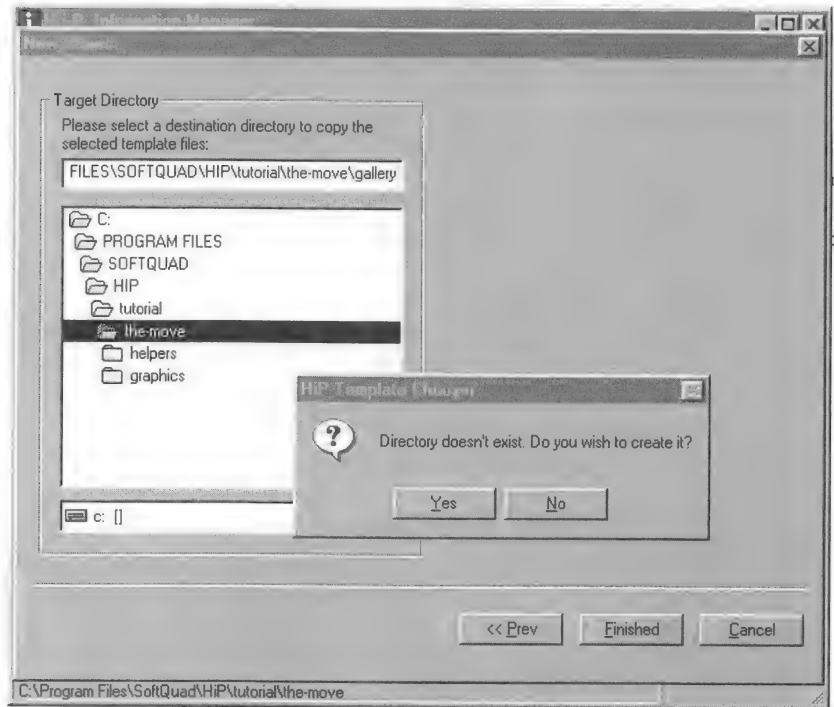
- Click on the ⌈ Yes ⌉ button when you are asked if you would like to create a new directory.

After a moment, H.i.P. will create a new project in your new directory. The Project display will will show only an *index.htm* file within your new project even though other files are created.

However, this new project is actually nested within *the-move*. Open *the-move* again and see where *gallery* fits in.

■ Choose *the-move* from the list of recently opened projects at the bottom of the **File** menu.

When *the-move* is reloaded, you will find a new folder, *gallery*, with a plus icon beside it.

■ Click on the ⊞ .

Now you can return to the *gallery* 'sub-project', and import *photgall.htm* with its various links.

■ Choose *gallery* from the list of recently opened projects at the bottom of the **File** menu. The full path would be something like:

```
C:\Program Files\SoftQuad\HiP\Help\
tutorial\the-move\gallery\
```

■ Choose **Import File...** from the **File** menu.
■ Click on the ⬚ Browse ⬚ button, and navigate to *the-move* (not *samples* this time), and select *photgall.htm*.
■ Click on ⬚ Open ⬚ .
■ Make sure your **Destination** is *gallery* (not *the-move* this time).
■ Click on ⬚ OK ⬚ .

Your Project display will look something like this:
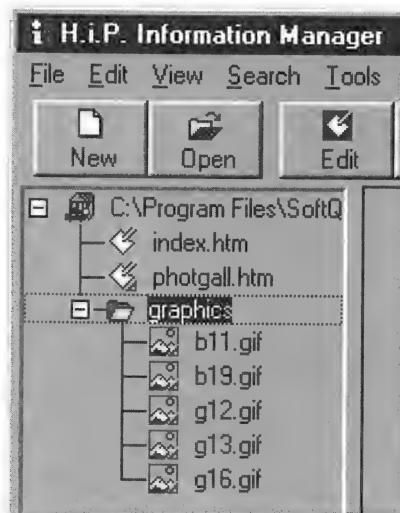
SoftQuad H.i.P. make automatic adjustments to link information during the import, including the automatic creation of a new folder, *graphics*, in the new project folder, *gallery*. When a *linked* file is moved, H.i.P. changes the links in all the files that link to that file to reflect its new location. Links within the moved file are also adjusted.

Hal would later choose the same technique for all of the documents that described how to prepare a piece of machinery for shipment. Equipment was constantly being acquired and sold, and individual machines were often shifted from one part of the plant to another. This information would persist and grow as the employees gained experience with each item of machinery. Your company probably has a similar group of 'How-To' documents that can acquire detail about complex procedures. If they are kept up-to-date, they can make an important contribution over the long term.

Before you move on to the next section, you should re-open your project, *the-move*.

■   Choose *the-move* from the list of recently-edited folders in the **File** menu.

Creating a new cascading style sheet

Just after Hal had finished moving documents into his new sub-projects he got a call from Yoshiaki, the foreman who had compiled the machine preparation information. He was as pleased as Hal with the way the UDE showed up in the Live TOC; he was able to do a context-sensitive find in his H.i.P. Viewer on all the details of the plant move that posed special problems.

However, Yoshiaki did have a suggestion for improving the use of the SpecialAttention UDE; what if they were to appear on the screen in a different color? Hal knew it was possible to achieve this effect for readers who were using Microsoft Internet Explorer, but not for those using Netscape Navigator. He had already used cascading style sheets to make two different H.i.P. views of the same file (see page 88) and he thought he might be ready for this next step.

The appropriate style was probably bold type, set with a wider margin, the way notes are often typeset in a book. But, when the information appears on a computer screen there was an additional opportunity—and a possible hazard. The SpecialAttention UDE could be presented in another color (and Hal favored red). On the other hand, he knew of at least one employee who was color-blind to red and green. This is were the word 'cascading' started to make some sense. Certain aspects of the style that Hal chose could be mandatory while others were optional. He wanted to insist on the wide margins and boldface, but also was willing to allow particular users to reject his choice of red in favor of their own choice for their local viewing. Hal used the cascading style sheet editor to create a style attached to the SpecialAttention UDE:

- Switch to the H.i.P. Information Manager with *the-move* open (if it isn't already).

- Choose **Styles...** from the **Tools** menu.

- Navigate to the *helpers* folder in *the-move* folder.

- Type a new filename, *specattn* into the text box. You will recall that there is another file with the name *specattn*—but it's an *.hpl* file that contains a Live TOC. The file you are creating now will be *specattn.css*.

- Click on the [Open] button, then click on the [Yes] button when you are prompted to create the new file.

The **CSS Editor** dialog is expandable. By clicking on the [More...] button, you can gain access to advanced cascading style sheet functionality such as dependencies between one style and another. One of these days, you may need to know about inter-dependencies and other aspects of the css standard, and when that time comes, please refer to the *Styles and Views* chapter of the H.*i*.P. *Reference Guide*. In the meantime, you can get a lot done by assigning style characteristics to elements and user-defined extensions.

If you were curious and clicked on the [More...] button:

■ Click on the [Less...] button now.

Your **CSS Editor** dialog should have **Title** and **Element** text boxes at the top.

■ Give your stylesheet a title. Type `Special Attention` in the **Title** text box.

■ Click on [Add UDEs...] and open the file *the-move.hpe*.

■ Select the **SpecialAttention** UDE from the **Element** list box.

We aren't finished yet, of course, but, before we go any further, note the **Hide Contents of Element** check box above the sample text on the right. That's the check box that Hal used when he created two different views for the *slspack.htm* packing instruction file. There was one view for the Priority people and another for Exemplar. In *priority.css*, he hid the Exemplar material, and vice versa for the *exemplar.css* styles file.

■ Click on the **Font** tab.

■ Choose **Bold** in the **font weight** list box.

■ Click on the **Text** tab.

■ Choose red in the **color** list box (at the bottom).

Note that the sample text has changed to boldface in red.
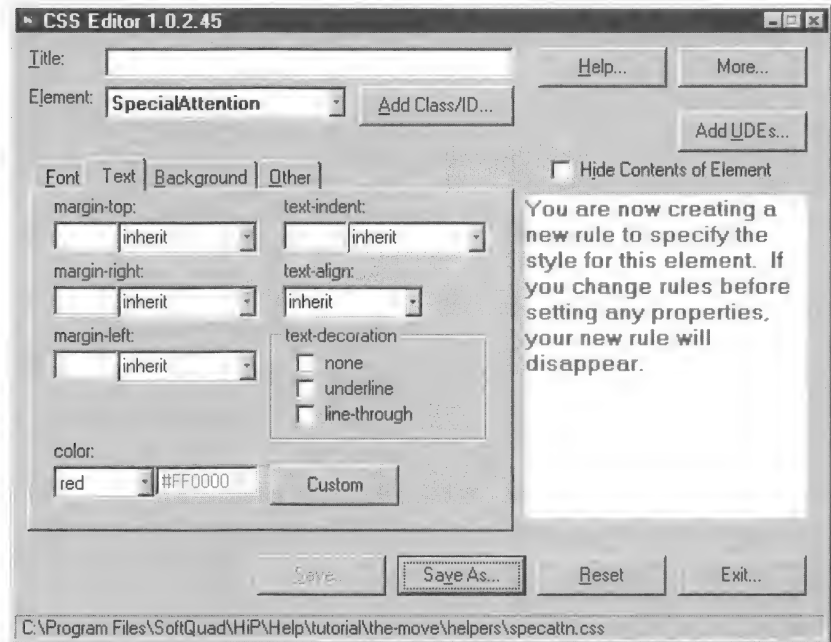
- Click on the [Save] button, then click on [Exit...] .

The next set of steps links the *equ-prep.htm* file to the *specattn.css* stylesheet.

- Switch to the H.i.P. Information Manager and choose *equ-prep.htm* for editing. (Right-click on the file name and choose **Edit File** from the menu.)
- Choose **Properties...** from the **File** menu.
- Click on the **Properties** tab.
- Click on **Views/Styles** and make sure that the title 'Special Attention' is displayed.

If it's not there:

- Click on the [Add...] button, select the *helpers* folder in your *the-move* folder, and click on the check box beside 'Special Attention'.

- Click on the OK button, then click the OK in the **Document Pro-perties** dialog.

You may want to confirm that a new Link element has been created.

- Choose **Show Head Element** from the **View** menu.
- Confirm that there is a new LINK element that contains a reference to the file *specattn.css*. It should have the following attributes:

  – REL="STYLESHEET"
  – TITLE="Special Attention"
  – HREF="../helpers/specattn.css"

Now you can view the file and see whether the text that is contained within the SpecialAttention UDE looks ... special.

- Save the file.
- Click on the Internet Explorer preview buttons on the toolbar.

Taking care of business

Exemplar Manufacturing's moving committee was busy planning and directing. Like any large committee they soon decided (unanimously) that a certain degree of formality was required in the conduct of their responsibilities: Paperwork with a capital 'P'! It began to arrive on Hal's desk after the first meeting. There were reports, then there were comments and revisions of reports; there were minutes and schedules; there were procedures documents and even approval forms for resources expended in aid of committee business (donut documentation).

Hal, an agreeable sort, was almost up to his knees before his epiphany arrived. He took a sheet from the mid-range of the stack and there, at the bottom was what he needed. Exemplar's managerial assistants always placed their word-processor file names in small type at the bottom of their paperwork—they were exemplary, after all. Hal thought that the file,*H:/projects/1204.56/planning/reports/prelim/approved/9610/donna/move/1510.doc* looked like an excellent candidate: Hal called Donna.

Within minutes of receiving the file Hal was back on the phone inviting Donna to see the results. She stopped by Hal's desk and checked it out: one click on 'Move committee report, Oct 15' and she was there. "Holy jumpin!" said Donna.

A miracle? No, a URL (universal resource locator). Of course, Hal knew that his file structure might eventually acquire the same kind of complexity (well... not quite the same). But Hal would never have to remember it. Thanks to the hyperlinking strategy of Web pages, things could have sensible names, no matter how complex their actual location.

The fact that Hal was able to reproduce the report so quickly in H.i.P. was due entirely to his new-found faith in templates. Committee reports are standard fare, so your template library has some handy examples. While Donna was filling out a 'P.O. requisition' for her own copy of H.i.P. (that new fella wasn't going to take control of her files!), Hal was doing a little exploring among the templates. Of course, he found a H.i.P. version of the same 'P.O. requisition' form that Donna was just then filling out—it was in the 'accounting' group.
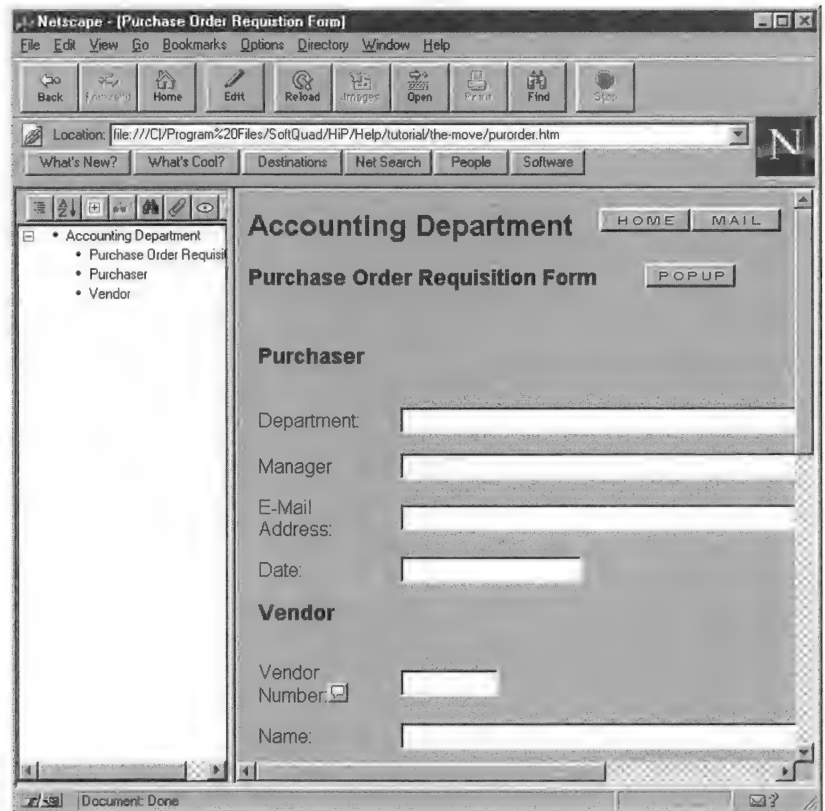
Would Donna relish the irony? No. Hal resisted the temptation and continued his own exploration.

You may want to do the same. Look through all of the template choices and keep them in mind as a starting point for your next page.

You can create minutes for your own meeting (or you can talk someone else into it) and present the result in your new site:

- Select the top line of *the-move* project (the one with the ⬛).

- Click on New File... in the File menu.

- Choose humanres in the template group.

- Select *minutes.htm*, and click on the ⌈ OK ⌉ button.

When the new file, *minutes.htm*, is created in your project, select it and view it.
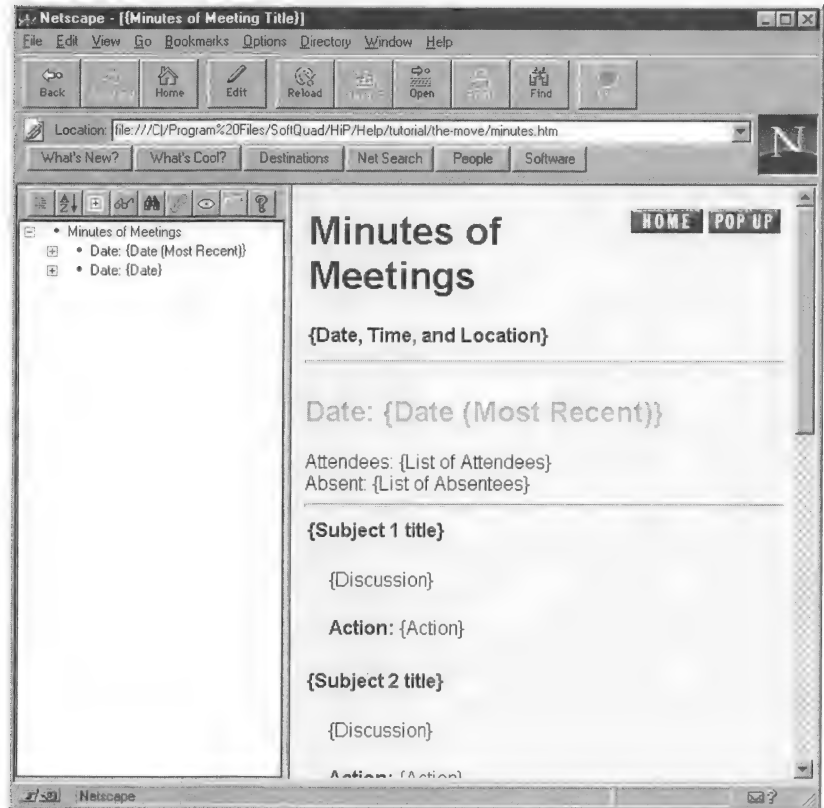
After looking over all the choices, Hal was intrigued by the idea of a H.*i*.P. calendar so he opened the template. It wasn't what he expected. In fact when he placed *calendar.htm* in his H.*i*.P. Viewer, he was taken aback, "It's ... alive!" he murmured.

But, before we get to Hal's encounter with the calendar template, take a quick look at all of the templates that are available to you and your H.*i*.P. contributors. Many of these will fill the bill exactly, but even if they don't, they are all filled with useful H.*i*.P. elements that will get you started on almost any common business form or Web page.

Time traveling with an active calendar

At first the calendar template looked like another monthly calendar—it just lay there. But, when Hal poked it with his mouse pointer, it started to move. He realized immediately that it was a three-dimensional calendar: it had columns and rows of days, but it also had views. Different people, choosing different views, would have different information about what was planned during the coming months. Often, the information would overlap. It could be one document, with one maintenance responsibility and one, unique and authoritative copy. Hal gripped his keyboard.

If you haven't had a poke at the calendar yourself, in the H.i.P. Information Manager:

- Select the top line of *the-move* project (the one with the ▓ .
- Choose **Import File...** in the **File** menu.
- Click on the [Browse...] button.
- Navigate to the *samples* folder in *tutorial* folder.
- Select *calendar.htm* and click on the [Open] button.
- Make sure your **Destination** is the *the-move* folder.
- Click on the [OK] button in the **Import File** dialog.

You will get an **Existing File** warning dialog because some of the 'helper' files that *calendar.htm* has links to were already imported when the folder *helpers* was created in the folder *the-move* earlier in the tutorial.

- Click on [Overwrite] because you haven't done any edits on these particular styles files.

Once you have imported the calendar, take a look at it in your H.i.P. Viewer.

- Highlight the file *calendar.htm*, and click on the ▓ button.

When the file has been loaded by the H.i.P. Viewer:

- Choose the **Production** view by clicking on the ▓ button and selecting the view from the menu.
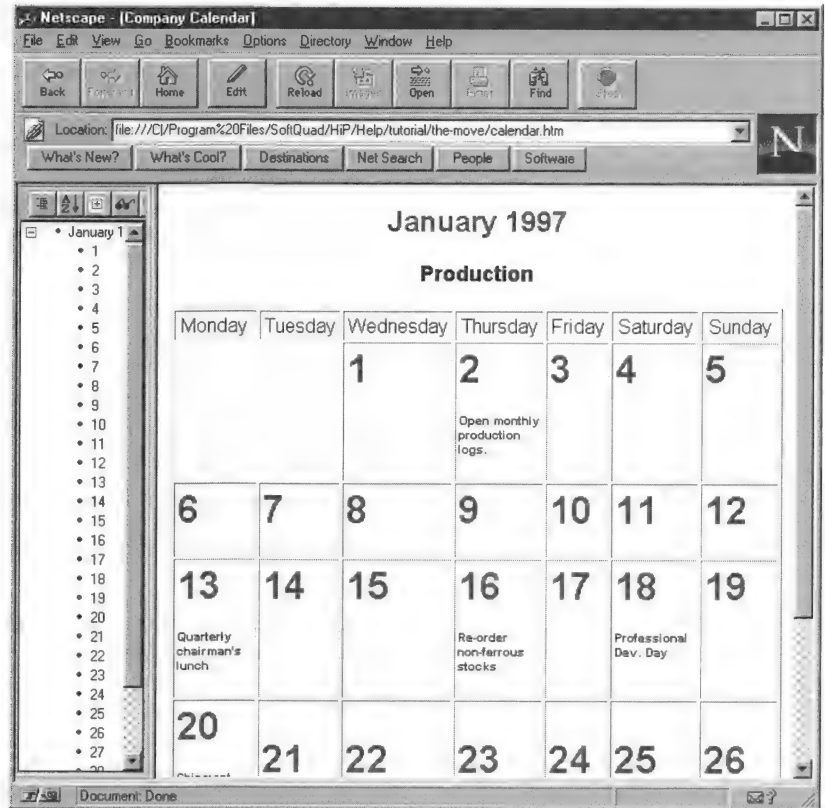
Naturally, Hal has organized the calendar so that it served the different schedules and interests of different departments, but also showed company-wide events. The sales people need a different view of things than the production people—and accounting has a third set of important dates. And some dates have to be visible to everyone. The technique Hal used was to invent four different CLASS attribute values:

- □ production
- □ sales
- □ accounting
- □ company-wide

To put a particular event in the correct category he surrounded it with an H6 element, then assigned the appropriate attribute value. Let's take an example:

- Switch to the H.i.P. Information Manager.

- Right-click on the file *calendar.htm* in the Project display and choose **Edit File** from the menu.

- Place your insertion point in the H6 element in the table cell that contains the text 'January 9' (it's the one about the Chinese delegation), and type ⌗F6⌗ .

- Notice that the **CLASS** attribute is set to 'company-wide'.

When the calendar is viewed, the H.i.P. Viewer checks the currently selected style sheet for rules applying to the class for each element. The rule for a particular class (more accurately, a **CLASS** attribute value) may indicate that it should be hidden.

You can add another event and have it display to the appropriate group:

- Place you cursor in the table cell for January 8, just after the ⟨/H5⟩ end-tag icon.

- Insert an H6 element and type `Brass forging stock shipped`, or some other important event.

- Type ⌗F6⌗ .
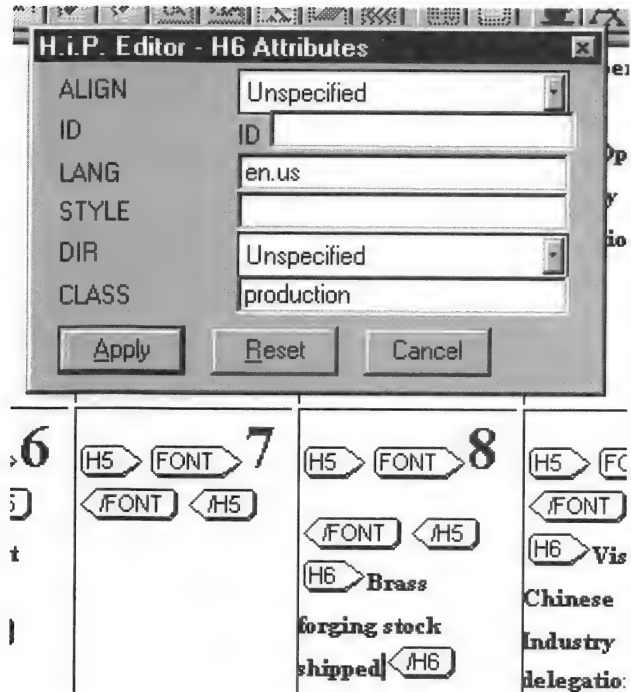
- Type production in the text box for **CLASS** (the lower case is important), and click on [Apply].
- Save and preview your file.

Since the calendar is a standard HTML table, you can add various elements in each cell—including links to documents that provide information regarding the day's events. A reader who clicks on a multiloc in the cell for Friday, January 3, could then read the conference schedule—and even fill out a form to attend.

Hal was so pleased with the calendar that he wanted to make sure everyone noticed. (He also wanted them to be properly informed, of course.) But, aside from phoning them up and telling them where to look, what could he do? It was the answer to this question that led to Hal's promotion from intranet editor to intranet publisher.

Getting the word out with subscriptions

The difference between and editor and a publisher, Hal discovered, is distribution. Having useful information, well arranged and attractively presented, is only half the battle. What he needed next was a way to make people aware that the information was available to be used—a kind of newsstand distribution and home delivery system for intranet documents.

Since the building that Exemplar was about to occupy was still under construction, the move committee wanted to keep track of the construction schedule for various parts of the new plant and offices. Hal was asked to create a construction progress page that would show a plan (view) of the new building and inform everyone when various sections would be complete. This would obviously affect the move dates themselves, which, in turn, would have serious consequences for production volumes in the plants. A simplified view of the plan is shown in the file *plan_01.htm*, 'Exemplar Plant'—you can import it from the *samples* folder in the *tutorial* folder.

- Switch to the H.i.P. Information Manager.
- Choose **Import File...** from the **File** menu.
- Click on ⎡ Browse ⎤ and navigate to the *samples* folder.
- Select *plan_01.htm* and click on ⎡ Open ⎤ .

Note that the files *sch-sale.htm* and *sch_stag.htm* tagged along when *plan_01.htm* was imported.

- Select *plan_01.htm* in the Project display and click on the Browser button .

When the file is loaded in the H.i.P. Viewer:

- Click on the area around the word 'Stock staging area' on the plant plan.
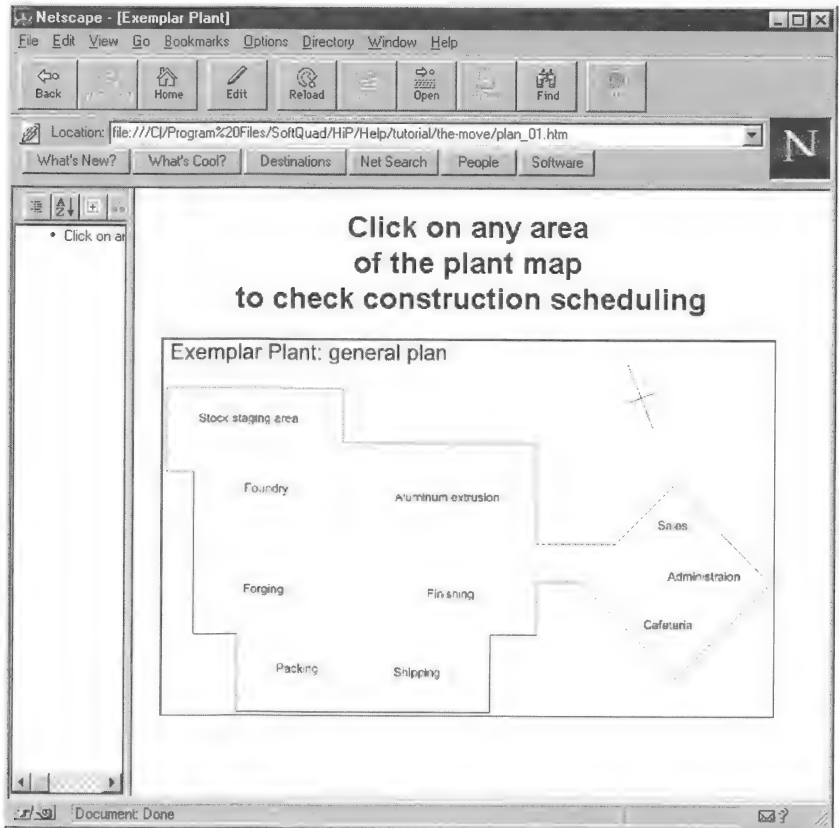
As you can see, Hal used image mapping to lead readers from the plan view to the actual schedule and construction news for each section of the plant. 'Mapping' and image with links is a fairly common Web page technique and your H.i.P. Editor will help you construct such an image map. If you would like to learn how, consult the *Images* chapter of the H.i.P. *Editor Reference Guide*.

While this would ensure that the construction schedule, with all of its progress and setbacks, would be available, it didn't mean that everyone on the planning committee would know it was there or had changed. Here was a problem with a built-in solution. Hal needed a distribution

system, so he took his first look at the H.i.P. Monitor and planned his next step.

The H.i.P. Monitor was running on Exemplar's Web server, not on Hal's own PC. Hal was a little surprised by the H.i.P. Monitor interface and you may be too—it has a browser interface. That means you talk to it, and it responds to you, in the H.i.P. Viewer (your Netscape or Internet Explorer browser with H.i.P. plug-ins). This is similar to filling out a form when you're viewing a Web page. Of course, the Monitor Web pages have some unique. Communicating with it was simple, though: Hal used his H.i.P. Viewer to surf over to the H.i.P. Monitor homepage.

The H.i.P. Monitor is like having your own foreign correspondent filing stories. It searches through the Web server and its operating system looking for events to report. When it finds something interesting, it sends e-mail.

At Exemplar, the system was run by Marco (you met him earlier in the tutorial). Marco was interested in things that happened in the operating system—disks filling up, the network going down, unauthorized logins—so he provided his own e-mail address to the agent for reporting on these kinds of events. But the H.i.P. Monitor can also determine and report on events such as Web pages expiring, links breaking, and projects being published. As the intranet publisher, Hal was responsible for keeping up-to-date on the integrity of the intranet content.

If you don't know your intranet's Monitor Home Page address:

- Switch to the desktop and open the SoftQuad H.i.P. Applications folder (unless you have installed the applications elsewhere).
- Click on the H.i.P. Monitor Administrator icon.
- Add the the Monitor Home Page URL to your bookmarks or favorites.

Among its many site administration functions, the H.i.P. Monitor helps keep readers informed by sending out e-mail notification of changes to pages that may interest them. There are three ways that the H.i.P. Monitor can be made aware of someone's particular interest in a page, a topic, or author of a page:

☐ The reader can subscribe from a H.i.P. Viewer.

☐ The reader's name can be added to a distribution list by a document's author.

☐ The reader's name can be added to a distribution list by a document's publisher.

The H.i.P. Monitor also allows a publisher to keep tabs on the integrity of links among project pages or to pages outside of the project. If you have information that will expire and need replacement, the H.i.P. Monitor can watch the computer's clock for you and make sure the information is kept current.

In this case, Hal was interested in how document subscription might work for his planning committee. He decided to subscribe himself to his own construction progress page. Then, every time he made an edit on the page, noting a delay, or announcing completion dates, he would receive a message on the company's e-mail system. He used his H.i.P. Viewer to surf to the construction progress page and sign himself up, then he used his H.i.P. Editor to open the construction progress page and see if the subscription would work. He changed the completion date for the concrete floor in the electrical room, then saved the page. A few moments later, a message arrived:

```
In response to your request for notification: the page

    http://exemplar.com/html/the-move/sch_stag.htm

has changed.

If you want to cancel your subscription to this author,
go to the following URL

    http://exemplar.com/H.i.P.Monitor/admin.cgi/
    unsubscribe?email=hmallard
```

Subscriptions were the answer. He created a distribution list comprising all the members of the committee. To see how to make a distribution list, see page 130. (Luckily, he also remembered to go back and correct the date on the construction progress page.) If you'd like to subscribe from your H.i.P. Viewer, here's what do to:

■ Surf to the page that interests you. It doesn't have to be the construction progress page; the H.i.P. Monitor will sign you up for whatever page interests you.

- Click on the subscribe button.
- Use the **Subscribe** dialog to specify whether you are interested in changes to **This page, Selected topics, Selected authors,** or all three.

The next time the page is changed, the H.*i*.P. Monitor will become aware of the change (it has monitoring agents that wait and watch constantly), then it will check its list to see who is interested and send an e-mail notification.

Getting published

The time had finally arrived. Hal had kept all this information to himself, but it was time to go 'live from the net'. That global publish button had been sitting up on the right side of his H.*i*.P. Information Manager just waiting for this moment. Hal took his mouse firmly in hand and clicked on the .

Here's what he saw:
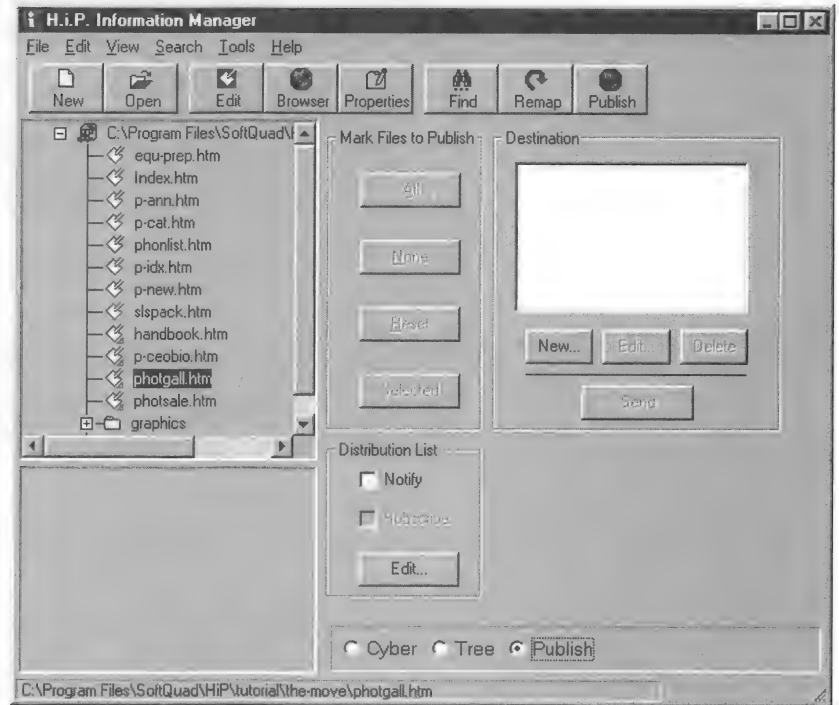
Publishing means copying a fully-prepared set of files to a site where they can be shared. Hal had been working on his PC's local drive and now it was time to copy 'The Move' up to Exemplar's intranet server.

The first choice to make was the name of the Web server to which 'The Move' would move.

■   Click on the [New...] button.

The **Destination Editor** dialog appears.
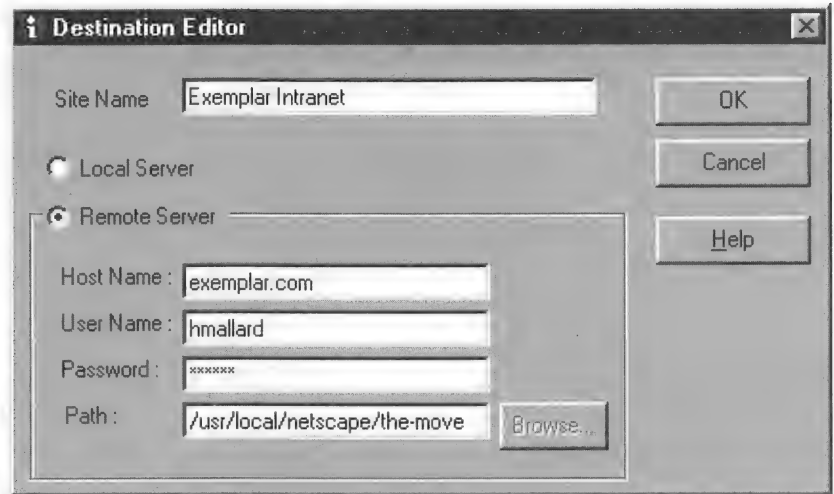
- Type the destination Web server's name in the **Site Name** text box.
- Click on either the **Local Server** or the **Remote Server** radio buttons.

If you are publishing to a local server, you won't need a user name or password, but you will need a path. This information will be available from your system administrator.

- Fill in the **Host, User** and **Password** information.
- Browse for the correct **Path.**
- Click on the [OK] button.
- Verify your password by typing it again and clicking on [OK] .

Note that you can publish to a location that's not on a server (it might be on your local drive) if you are just trying things out. You may prefer to take this step as a trial run before you go public.

Once you have specified a destination (and there can be more than one):

- Select the destination(s) in the **Destination** list. The buttons in the **Mark Files to Publish** section will become active.

Now you can choose the files that are ready to publish. Under the **Mark Files to Publish** section of the dialog, you can choose **All**, **None** or **Selected**— this last choice refers to files that you may have highlighted in your Project display. If you've made an error, click on [Reset] . The files

that are chosen for publication will be marked in the Project display with a check mark. The default choice is **All Files**.
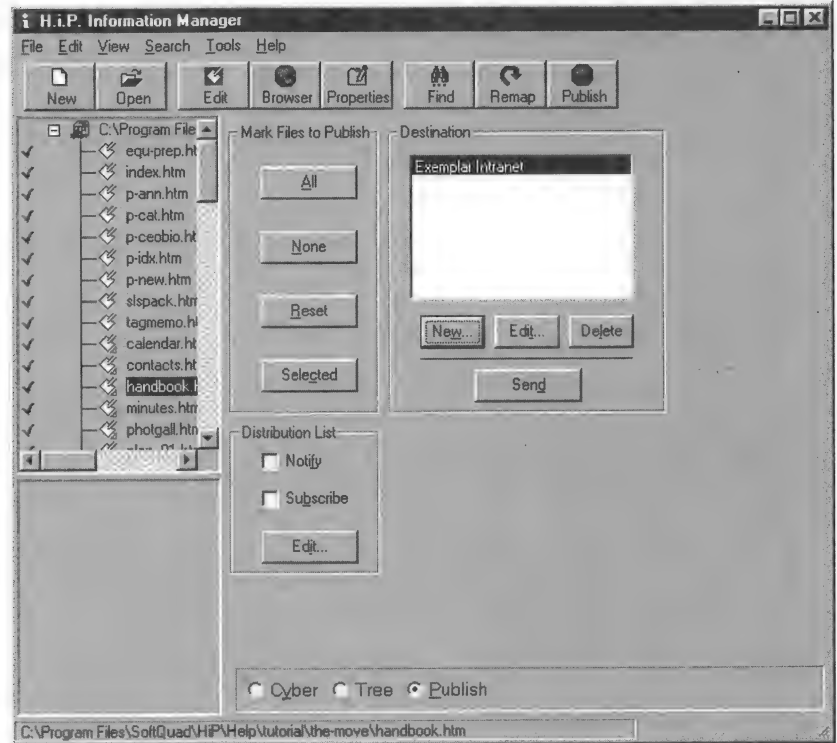
If you are ready, click on ⌈ Send ⌉ .

Notifying your readers with distributions

Allowing readers to 'pull' information from the server with individual subscriptions made from their H.i.P. Viewers can be effective. But Hal knew that some people would have to be alerted to important information updates whether they were subscribed or not. This 'push' to readers (that is, notifying them actively) can be accomplished in two ways:

☐ Distributions are lists of e-mail addresses. One notification will be sent to each of these addresses upon publication of either a site or part of a site.

☐ Subscriptions are also lists of e-mail addresses. These are subscriptions that you 'give' to a user, rather than waiting for the user to sign up.

Hal planned to use a distribution to make sure that everyone on the Move Committee was at least notified of changes to the site every time he published. That way, nobody could complain that they weren't informed. A distribution list is stored with the H.i.P. document and is used by the H.i.P. Monitor to prepare and send notification. Hal created a sample list just to test a distribution.

We will assume that you are looking at the Publish dialog in the H.i.P. Information Manager.(If not, see page 126.) Distribution lists are created as part of the publishing process:

■ Click on the [Edit] button. The Distribution List Editor will appear.

■ Click on [New...] . The Address Book Edit dialog will appear.
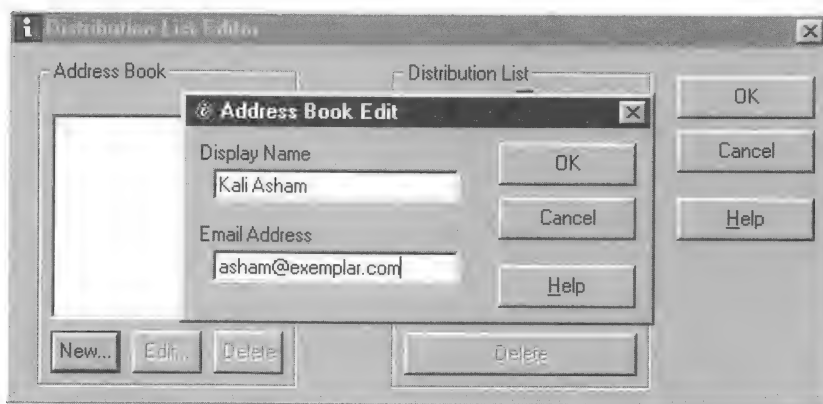
- Type in a user's name—Anthony LaSalle, for example—and e-mail address.
- Click on the [OK] button.

To place names on a distribution list:

- Select the person's name on the left side.
- Click the [>>] button.

If you need to delete a name from the **Distribution List** box:

- Select the name to be deleted and click on the [Delete] button at the bottom of the distribution list.

If you want to delete the name from the **Address Book**:

- Click on the [Delete] button.

When you are finished editing your lists:

- Click on the [OK] button.

After you have created your list, you can specify whether to **Notify**, or both **Notify** and **Subscribe** the people on your list.

- Click on the [OK] button when your list is complete.

When you are ready for take-off:

- Click on the [Send] button.

You've been published. Whenever you change a page or re-published it, everybody on your list will sit up an take notice.

Automatic updates
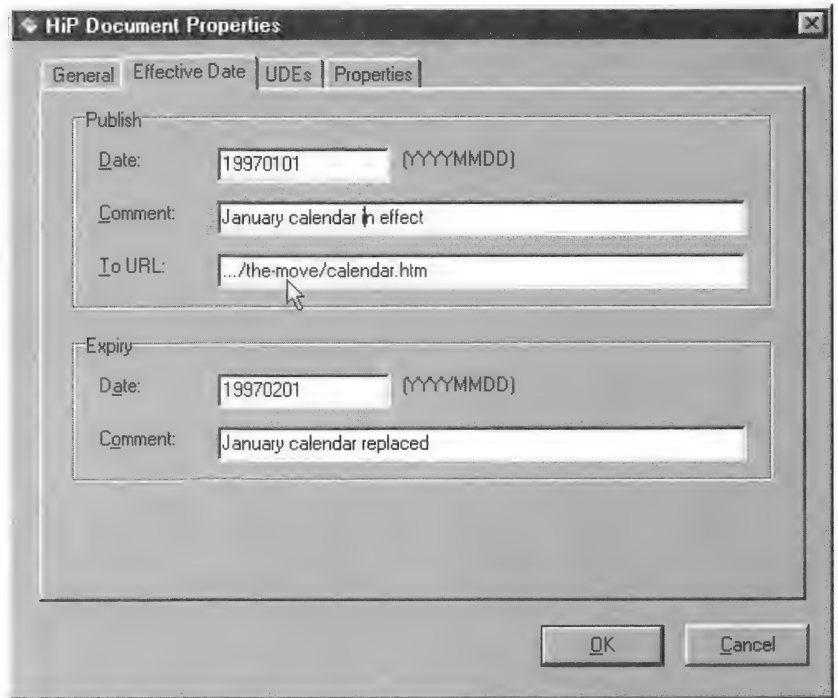
The Calendar was still Hal's favorite, though he had to admit that keeping it up-to-date was going to be a daily chore. Life would be a little easier, he decided, if he could find a way to have the calendar page help with the housekeeping. One possibility would be to have the page renew itself daily. Even though it was a monthly calendar, lots of events popped up during the month. Here, the H.i.P. Monitor Administrator was helpful.

Hal's plan was to add events to the calendar page every day then publish the page with an effective date, expiry date and the name of the new version of the file that should replace it. Hal was able to prepare the calendar files a day or two in advance and be certain that they would take effect on time, then expire and be removed when their replacements came into effect. You can add a couple of events to your calendar template and see how this might work:

■ Switch to H.i.P. Information Manager and right-click on *calendar.htm* in the Project display.

■ Choose **Edit File** from the menu.

When the file has loaded into the H.i.P. Editor:

■ Choose **Properties** ... from the **File** menu.

■ Click on the **Effective Dates** tab.

■ Fill in the date fields. This (January) document should be effective from Jan 1 to Jan 31. Write the date in the YYYYMMDD format—that means: a four-digit year, a two-digit month and a two-digit day, in year-month-day order.

- Write a comment. This message will appear in an e-mail message when the page become effective or expires.
- Type in the name of the file that this file will replace.
- Click on the [OK] button.

Once this information it made available to the H.i.P. Monitor Administrator, the Monitor Administrator will take care of the rest. However, before the H.i.P. Monitor can read and act on such 'meta data', the page has to be published (see page 126).

Time to get a move on

When it was finally Hal's turn to pack up his office and move, he re-read his packing instructions, made special arrangements for his office plants, and checked the new location's floor plan for the exact position of his new work station. Then he remembered. He never did check to see whether the town of Neoteric had a bowling league. Or a model railroad shop. Or a park with a duck pond. All of that information could easily have filled a whole sub-project of 'The Move'. Ah well, there was plenty of opportunity to try out new ideas in H.i.P.—in fact he was planning to get started on Exemplar's production management intranet as soon he was settled in the new place. Imagine putting the production schedules for the whole manufacturing plant on a H.i.P. calendar…

H.i.P. Editor Tutorial

If you're new to creating intranet or Web (HTML) documents, you may want to work through these short tutorials.

The tutorials cover the following topics:

- The basic document: titles, headers, and paragraphs (the next page)
- Character formatting: inline text (page 141)
- Block formatting (page 142)
- Lists: ordered lists, unordered lists, and definition lists (page 143)
- Links and URLs: anchors and images (page 147)
- Backgrounds, colors, and fonts (page 154)
- Alignment (page 158)
- Finding and replacing URLs (page 159)
- Forms (page 161)

The tutorials don't cover each topic exhaustively. If you need more information, consult the chapters *Core* HTML and *Extensions to* HTML. Both are in the H.i.P. Editor section of the Reference Guide, which gives you details on every aspect of editing documents with SoftQuad H.i.P. .

The first tutorial can be used as a 'quick start' that shows you how to create a file in the H.i.P. Editor component of HoTMetaL PRO Intranet Publisher. When you've finished that section, you can continue with the

other sections or, if you feel comfortable using or experimenting with HTML, you can skip over the rest of the tutorials and refer to the Reference Guide when you need to find out something about HTML.

If you have been reading the H.i.P. Tutorial, you will have met Hal, recently-appointed publisher of Exemplar Manufacturing's first intranet site. Hal felt he needed a little extra practice with the H.i.P. Editor, so he tried his hand at a fairly standard corporate Web document—a laudatory biography of the now-former CEO of the now-defunct Priority Machining. Just for a change, Hal thought that he might tell the truth about Mike Woodhead—especially since the former President and CEO was enjoying his early retirement, and this practice document, Hal was sure, would never see the light of day. So let's have an editing workout with the life story of a corporate captain.

Getting started: a basic document

- Start up the H.i.P. Editor.

Four toolbars should be visible. If they aren't:

- Choose **Toolbars...** in the **View** menu.
- Turn on all four check boxes.
- Click on ⌑ OK ⌑.

Now:

- Choose **Open...** from the **File** menu.
- In the dialog box that appears, navigate to the *tutorial* folder inside the *Help* folder.
- Select the document called *tutor.htm*.
- Click on the ⌑ OK ⌑ button.

The H.i.P. Editor opens this document, which you can now edit. You should see small tag icons in the document, for example, $\boxed{HTML}$. If you don't:

- Click on the ▦ (show tags) toolbar button to make the tags visible. (If you click on this button again, the tags will become hidden. This button corresponds to the command **Show (Hide) Tags** in the **View** menu.)

This template contains all the required parts for a valid HTML document. It has a **TITLE** that you can edit:

- Click on the ▦ button.
- Click on the **General** tab.
- Type your title, for example, The REAL Story in the **Title** text box.

This is how the document will look in the H.i.P. Editor if you choose **Show Head Element** in the **View** menu.
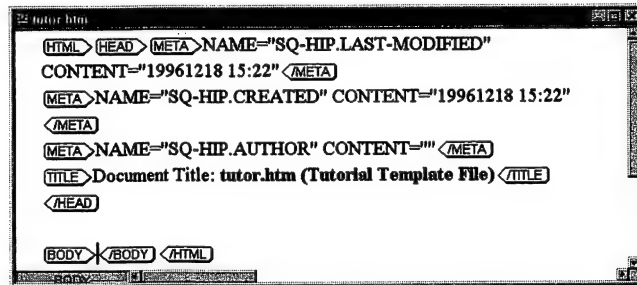
When you display this document in a browser, the contents of the **TITLE** element will generally be displayed in the title bar. When you display the document in the H.i.P. Information Manager, the title of the file will appear in the cyberbolic display (provided you have chosen **Show Titles** in the cyberbolic display's pop-up menu).

The main part of your document is the body—contained in the **BODY** element.

- Put the insertion point inside the **BODY** element (just to the *right* of the $\boxed{BODY}$ start-tag).

You have many choices of elements to insert. However, it's normal to start your document with a heading. HTML documents have six levels of headings, represented by the elements H1 through H6.

An H1 heading should be used for major divisions in your document.

■ Click on the [H1] toolbar button.

The H.i.P. Editor inserts an H1 element.

■ Type the following (or some other text of your choice) inside the H1 element:

    Mike Woodhead, ex-CEO Priority Machining Inc.

Now you're ready to insert some text.

■ Move the insertion point to the right of the ⟨/H1⟩ end-tag.

■ Click on the [¶] (paragraph) toolbar button.

■ Type some text such as the following:

> ⓅMike Woodhead, ex-CEO of Priority Machining. A story that strains credulity. Twice runner-up in the Puddle Duck Golf Tournament, winner of the coveted Medal for Contributions to Mixology, and a man with a grasp of business that employees refered to as "stunning", "unbelievable", and "tenuous." He may no longer lead us, but he shines as an example to us all. ⟨Ⓟ⟩

You can begin smaller subdivisions of the document with lower-level (H2 through H6) headings. You can skip levels if you want, but your documents will usually look better if you don't.

■ Move the insertion point to the right of the ⟨/P⟩ end-tag.

■ Click on the [H2] toolbar button.

■ Type the text: Biography

■ Move the insertion point to the right of the ⟨/H2⟩ end-tag.

■ Click on the [¶] toolbar button.

■ Type the text:

(P)Mike Woodhead first gained prominence in the metal forming industry when he went to work for his father-in-law, P.J. Blockhammer, founder of Priority Machining Inc. Mike's courtship of PJ's youngest and favorite daughter Lilly was a whirl-wind affair -- "right down to the wire," PJ was heard to remark at the time. As Vice President, Marketing, Mike soon discovered his true talent for business. His affability as a coach and mentor of several Fortune 500 executives not only improved their swings, but led to the unprecedented growth of Priority's order book. Mike was fond of saying: (/P)

The document should now look something like this:

(HTML)(BODY)(H1)**Mike Woodhead, ex-CEO Priority Machining Inc.** (/H1)

(P)Mike Woodhead, ex-CEO of Priority Machining. A story that strains credulity. Twice runner-up in the Puddle Duck Golf Tournament, winner of the coveted Medal for Contributions to Mixology, and a man with a grasp of business that employees refered to as "stunning", "unbelievable", and "tenuous." He may no longer lead us, but he shines as an example to us all. (/P)

(H2)**Biography**(/H2)

(P)Mike Woodhead first gained prominence in the metal forming industry when he went to work for his father-in-law, P.J. Blockhammer, founder of Priority Machining Inc. Mike's courtship of PJ's youngest and favorite daughter Lilly was a whirl-wind affair -- "right down to the wire," PJ was heard to remark at the time. As Vice President, Marketing, Mike soon discovered his true talent for business. His affability as a coach and mentor of several Fortune 500 executives not only improved their swings, but led to the unprecedented growth of Priority's order book. Mike was fond of saying: (/P)

Now perhaps you'd like to see what this document will look like when it's published on the Web. But first, save the file:

■ Click on the ▨ toolbar button, choose **Save** from the **File** menu, or type (Ctrl-S) at the keyboard, and save the file using the file chooser dialog box that appears.

■ Now, choose one of the four preview toolbar buttons, found on the right side of the third toolbar ('Other HTML').

The toolbar button will be filled with the icons of the browsers that you chose when you installed SoftQuad H.*i*.P. . They will be blank if you have never chosen a browser for that button. If you click on a blank button, you'll get a file chooser dialog box that lets you locate and choose a Web browsera dialog in which you can choose a browser will appear. When you have chosen a browser, the blank toolbar button will contain an iconfor the particular browser. The tooltip that is displayed when you hold the mouse cursor over the button will also tell you which browser is associated with that button. All future previewing can be done with one click on that toolbar button.

> **Note** Only the two browsers with H.i.P. Viewer plug-ins—
> Netscape Navigator and Microsoft Internet Explorer—can
> view H.i.P. documents with all their H.i.P. extensions.

To change the browser associated with a toolbar button, you must choose the 'Preview...' command from the File menu and delete the browser from the browser list. A toolbar button will become blank; a new browser can be selected either from the 'Preview...' dialog box, or by clicking on the blank toolbar button.

You can also preview documents without using the toolbar.

- Choose **Preview...** from the **File** menu, or type ⟨ Ctrl-M ⟩ at the keyboard.

If the list of browsers is blank:

- Click on the ⟨ Add... ⟩ button. This will give you a **Choose Browser** dialogfile chooser dialog box in which you can locate and choose a browser.
- Navigate to your browser program file and click on ⟨ Open ⟩ .
- Select a browser from the list and click on the ⟨ Preview ⟩ button.

The browser is launched, displaying your document.

See the *Previewing your file in a browser* section of the *Reference Guide* if you want more information on the **Preview...** command.

As we suggested at the start of the chapter, you may now wish to skip directly to the *Core* HTML chapter in the Reference Guide, or continue with the tutorial.

Character formatting: adding emphasis to inline text

This section is about formatting *inline* text—text that's embedded in a paragraph or some other block of text. Formatting blocks is covered in the next section. In an HTML document, you add emphasis to a piece of text by surrounding it with an element. This is similar to the approach of many desktop publishing programs, in which you would, for example, highlight a piece of text and choose a type style (such as bold or italic) from a menu.

- Switch back to the H.i.P. Editor.
- In the document you just created, highlight the words 'P.J. Block-hammer'.
- Click on the toolbar button.

This time, clicking on the toolbar button has *surrounded* the selection (with EM and /EM tag icons). The text should be formatted in italic (unless someone has changed the styles). In any case, a browser will generally format it in italic—you can use Preview... to try this out. Of course, you don't always have to surround the text *after* you've typed it—you can insert an EM element by clicking on the toolbar button when there's no highlighted text, and then just type the text between the tags.

You can also insert an EM element by choosing Emphasis from the Format menu.

For more information on character formatting, see the section *Character formatting* in the *Core HTML* chapter of the Reference Guide.

Formatting blocks of text

The most common way to mark up a block of text is with a P (paragraph) element, which you've already seen. There are several other elements that you can use to surround parts of your document that require special formatting. For example, suppose you wish to add a block quote to the sample document you created above:

- Move the insertion point to the right of the last ⟨ /P ⟩ end-tag.

- Click on the 📄 toolbar button.

This inserts a BLOCKQUOTE element.

- Type the letter I.

As soon as you do this, the H.i.P. Editor inserts a P element inside the BLOCKQUOTE. This is because the HTML rules do not allow text directly inside BLOCKQUOTE, but BLOCKQUOTE can contain P elements, which can contain text.

- Continue typing the text:

```
I can sell anything to those duffers. I
sell 'em before we make 'em!
```

Notice that the text is indented slightly to set off the quotation. A browser will display a block quote with similar special formatting.

For more information on block formatting, see the section *Block formatting* in the *Core HTML* chapter of the Reference Guide. To finish off this section, move your insertion point outside the ⟨/BLOCKQUOTE⟩ end-tag, insert a paragraph, and type the following:

```
To which PJ often added, "...before we even know
how to make 'em."
```

Insert another paragraph and type the following:

```
PJ's untimely death, at the age of 96, was a blow
to the entire company. Mike, however, rallied early,
transferring Lilly's shareholdings to an offshore trust
which later resold at a favorable price, placing majority
control in Mike's relaxed but firm grip. Within only
two years, Mike had transformed the predictable
Priority Machining into a throbbing headache for
its staff and shareholders. Notable among Mike's
financial techniques was the accounting of
```

```
revenue from sales of goods that were later
returned for reimbursement by unsuccessful distributors.
Mike's efforts went largely unnoticed until the telephone
lines were disconnected. Then, of course, industry
pundits pointed out the folly of attempting to compete
against cheap offshore labor.
```

You will practise doing some formatting on this paragraph later in the tutorial.

Lists

You can insert five different types of lists in your document. The three most common types of lists will be demonstrated in this section.

See the section *List elements* in the *Core HTML* chapter of the Reference Guide for more information on lists.

Ordered (numbered) lists

Ordered lists are lists with numbered items. You don't have to add the numbers yourself—a browser will add these for you (and the H.i.P. Editor will show them to you as well).

- Move the insertion point to the right of the ⟨ /P ⟩ end-tag in the sample document.
- Insert an H2 element from the toolbar.
- Type:
    ```
    Contributions of Mike Woodhead
    ```
- Move the insertion point to the right of the ⟨ /H2 ⟩ end-tag.
- Click on the ▨ (ordered list) toolbar button.

When you do this, the H.i.P. Editor automatically inserts an LI (list item) element inside the OL, unless you have turned rules checking off. (**Turn Rules Checking On/Off** is in the **Special** menu.) With one exception, all lists consist of one or more LIs.

- Inside the LI element, type:
    ```
    Increased company's store of golf jokes
    ```

Notice that the number '1' is shown before this list item. The H.*i*.P. Editor will display the numbers of items in an OL.

This is a convenient place to introduce a useful technique for adding elements: *splitting* the current element.

- Make sure the insertion point is just to the *left* of the ⟨ /LI ⟩ end-tag.
- Type ⟨ Return ⟩ or ⟨ Enter ⟩ .

The H.*i*.P. Editor splits the LI element, in effect creating a new, empty LI element after the current one. Splitting an element just before the end-tag is a convenient way of inserting a new element that has the same type as the current element. If you split an element in the middle of the text, everything before the insertion point goes in one element, and everything after goes in the other one. Another way to split an element is to choose **Split Element** from the **Markup** menu, or type ⟨ Ctrl-P ⟩ at the keyboard.

- Inside the new LI, type:
  ```
  Instituted 'casual
  Fridays'
  ```
- Split the current LI element.
- Type:
  ```
  Quality of water-cooler gossip improved
  ```
- Move the insertion point to the *right* of the last ⟨ /LI ⟩ end-tag.
- Type:
  ```
  Renovated CEO office
  ```

As soon as you start typing, the H.*i*.P. Editor inserts an LI element, because you can't type directly inside an OL. In general, if you type somewhere that text is not allowed, the H.*i*.P. Editor will attempt to insert a valid element so that you can continue entering text.

- Continue typing.

To see how ordered lists look, you should preview the document:

- Save the file.
- Click on one of the preview toolbar buttons, choose **Preview...** from the **File** menu, or type ⟨ Ctrl-M ⟩ at the keyboard.

As you can see, the browser has inserted the list numbers automatically.

Unordered lists

An unordered list is one whose items aren't numbered, but instead start with 'bullets' of some kind. Actually there are several kinds of unordered lists available to you; here we'll use the most common and general-purpose list element, UL (unordered list). In this section you'll also learn a new markup feature: changing the element type.

- Move the insertion point to a position between the (OL⟩ start-tag and the first (LI⟩ start-tag.

- Click on the ▤ (unordered list) button in the toolbar.

The OL start- and end-tags have changed to UL tags. If you click on a toolbar button, and theH.i.P. Editor can't validly insert an element, it'll try to change the current element to the one you've chosen. In this case, the only thing an OL can change to is a UL. (You can also change the current element by choosing **Change Element...** from the **Markup** menu, or typing (Ctrl-L) at the keyboard.)

Notice that a bullet is shown before every list item. The H.i.P. Editor will display the bullets before each list item in a UL.

- Save the file.

- Click on one of the preview toolbar buttons, choose **Preview...** from the **File** menu, or type (Ctrl-M) at the keyboard.

The browser now displays the list items with bullets rather than numbers.

You can *nest* lists by inserting a UL, OL, etc., inside a list item (LI). Some browsers—and the H.i.P. Editor—will change the list bullets for a nested list.

The other unordered list elements are MENU and DIR—see the section *List elements* in the *Core HTML* chapter in the Reference Guide for more information.

Definition lists

A third type of list is the 'definition list', or 'glossary list' (DL). This list is different from the others because it doesn't consist of list item (LI) elements. Instead, it consists of a series of terms (DT) and definitions (DD). As the name implies, this list is intended to display definitions, though you can use it for any list that requires two-part entries. When a definition list is displayed in a browser, a definition is typically displayed on a separate line from the corresponding term, and indented slightly.

- Insert a new P element after the last list.
- Type:

  ```
  Mike Woodhead used to talk in a dialect
  that his employees referred to as 'Mikespeak'.
  Here is a brief translation of some common
  Mikespeak terms:
  ```

- Click on the ▦ (definition list) toolbar button.

Notice that the H.i.P. Editor inserts the DL element just after the paragraph, because lists can't be inserted inside paragraphs.

The valid elements inside a DL are DT (definition term) and DD (definition). Although you're allowed to arrange them however you like, the normal ordering would be to have one DT (or more, if, for example, you are defining several words with the same meaning), followed by one or more DDs.

- Click on the ▦ (definition term) toolbar button. (If you were to just start typing here, instead of inserting a DT from the toolbar, The H.i.P. Editor would insert the DT for you, because you can't type directly inside a DL. As you become more familiar with the HTML rules, you can take advantage of shortcuts such as this.)
- Type:

  ```
  You're about as useful as a bent putter!
  ```

- Move the insertion point to the right of the ⟨/DT⟩ end-tag.
- Click on the ▦ (definition) toolbar button.
- Type:

  ```
  Your work is not to my satisfaction.
  ```

Now create two more definitions:

- Insert a DT element after the DD.
- Type:

    I'm going for an important sales meeting.

- After the DT, insert a DD.
- Type:

    You can find me on the golf course.

- Insert a DT element after the last DD.
- Type:

    I make Bloody Marys with more punch than you!

- After the DT, insert a DD.
- Type:

    Your work is not to my satisfaction.

To see how this list is displayed in the browser:

- Save the file.
- Click on one of the preview toolbar buttons, choose **Preview...** from the **File** menu, or type (Ctrl-M) at the keyboard.

Notice that the definitions are set off from the terms.

Anchors and images

It is normal for HTML documents to contain links to other documents, which can be located anywhere on the www. These links are provided by URLs (*Uniform Resource Locators*), which name the location and filename of a document, and the method (scheme) used to access it.

Anchors

When you want to create 'hot text' that someone can click on in a browser, causing a document to be accessed, you should use an 'anchor' (A) element.

- Move the insertion point just to the left of the ⟨/BODY⟩ end-tag.

- Type the text:

  See also the

  (The H.*i*.P. Editor will surround the text with a paragraph.)

You can insert an anchor in the H.*i*.P. Editor by using keyboard, menu, and toolbar commands , or by dragging and dropping. We will first walk through the process of creating an anchor within the H.*i*.P. Editor, and then go through using Windows drag and drop to create an anchor.

- Click on the 🔳 toolbar button. (This is equivalent to choosing the **Insert Anchor...** command in the **Markup** menu.)

This creates an A element, and brings up the **Edit URL** dialog box.

Next to the **Scheme** label in this dialog box there are a text box and a drop-down list that let you choose a *scheme*, which describes how the file referred to in the URL will be accessed by a Web browser. For example, if the document were on a Web server, you would choose the scheme *http*. In this exercise, you will use the *file* scheme. This scheme is used if the file being referred to is on your local file system. You need not insert the *file* scheme here; the H.*i*.P. Editor will do it for you.

- Click on the [Choose File...] button.

This causes a file chooser dialog box to appear.

The file you should choose is located in the folder *doc* in the the H.*i*.P. Editor installation folder.

- Navigate to that folder and open the file *works.htm*.

Notice that the file name and folder path have been inserted in the text box next to the [Choose File...] button, and *file* has been inserted in the scheme dialog box. You could have typed the file name there directly, and chosen the *file* scheme manually, but it's probably more convenient for you to use the file chooser dialog box.

Most of the time URLs will also have a *host* component, specifying the Web server of the file that's being pointed to, but in this case you don't need one because the file is on your local system.

- Click on the [OK] button.

If the mouse pointer is positioned between the start- and end-tags of the A element, the URL that you entered will be displayed in the message area in the lower left corner of the H.i.P. Editor window. If you've chosen **Show URLs** in the **View** menu, the URL will also be displayed to the right of the (A > start-tag in the document window.

Anchor elements can also be inserted by dragging and dropping external files into the document. This will create an anchor element with a link to the file.

The anchor that you will now create by dragging and dropping is identical to the anchor that you just created by inserting the A, so you may want to delete that anchor before creating this one.

- Switch to the Windows desktop or Windows Explorer and find the file *works.htm* in the *doc* folder under the H.i.P. Editor folder.

- Move and resize the document and application windows so that you can see both the H.i.P. Editor window and the *works.htm* file.

- Select the file *works.htm* by clicking and holding the left mouse button down on the file's icon.

- Continuing to hold down the left mouse button, move your mouse cursor to the H.i.P. Editor window. An image of the file you are dragging will appear under the mouse cursor. As the cursor is moved to the H.i.P. Editor window, it becomes a special drag and drop copy cursor. If you drag within a block of text, an insertion bar will appear behind the cursor.

- Release the left mouse button wherever you want the anchor element to appear.

When you release the mouse button, an A element will be created; its URL will point to the file that you just dragged. If you release the left mouse button at a point where an anchor would be invalid, nothing will happen. You'll have to drag and drop the file again to a valid location.

Note If you drag and drop to a saved document, the URL of the anchor that has been created will be in 'relative' format. If you drag and drop to an unsaved file, or create an anchor using 'Insert Element' or the anchor toolbar button, the anchor will have a 'complete' URL. See the Links chapter of the Reference Guide for more details on relative and complete URLs.

The next step is to enter the 'hot text', a phrase that the user will click on in their browser to access the file pointed to by the URL.

- Inside the A, type the text:

```
Financial Bugle
article,'The Fall of Mike Woodhead'.
```

This is the hot text. Most browsers will display it in a different color from surrounding text, and perhaps underlined, to alert the user that it points to another document or location.

To see how this works, you should view the document in a browser:

- Save the file.
- Click on one of the preview toolbar buttons, choose **Preview...** from the **File** menu, or type (Ctrl-M) at the keyboard.
- In the browser, click on the words "Financial Bugle article, 'The Fall of Mike Woodhead'."

The browser will now display the file *works.htm*.

Links to a specific location

It's possible to make a link to a specific location in the same document or in another document. Then, when you click on an anchor (call it the 'source') the browser window will display another anchor (the 'target') at the location you've linked to, opening another document if necessary.

- Scroll back toward the top of your document.
- Highlight the word 'Biography' in the first H2 element.

This piece of text will be the target anchor.

- Click on the ![toolbar button] toolbar button, or choose **Name Target...** from the **Markup** menu.

The dialog box that appears lets you assign a *name* to the target anchor. The browser will use this name to locate the anchor. The first word in the document's selection becomes the default name, but you can enter other text if you wish.

- In the **Name** text box, type:

  ` Bio`

- Click on the [OK] button.

Notice that the highlighted text is now surrounded by an **A** element.

You have created the 'target' anchor. Now you have to set up the 'source' anchor:

- Scroll to the bottom of your document.
- Move the insertion point to the left of the ⟨/BODY⟩ end-tag.
- Insert a **P** element.
- In the new **P** element, type:

  `Return to the 'Biography'`

- Highlight the word 'Biography'.
- Click on the 🔗 (connect link) toolbar button, or choose **Connect Link** from the **Markup** menu.

The highlighted text is now surrounded by an **A** element. If the *mouse* cursor is positioned between the start- and end-tags of the **A** element, the message area in the lower left corner of the H.i.P. Editor window reads **#Bio**.

If you position your insertion point within the **A** element near the top of the document (the one that contains the word 'Biography'), the message area will read **Name: Bio**.

The H.i.P. Editor has created a source anchor whose URL refers to the name 'Bio' that you just gave the target anchor.

Now you're ready to see the effect of what you've just done.

- Save the file.
- Click on one of the preview toolbar buttons, choose **Preview...** from the **File** menu, or type [Ctrl-M] at the keyboard.

- If both of the anchors you just created are visible, resize the browser window so that you can see only the one at the end of the document.
- In the browser, click on the words 'Biography'.

The browser window will now scroll so that the location of the target anchor is visible.

You can also use these commands to create a link from one document to a specific location in another document. Once you create the target anchor, the H.i.P. Editor remembers the name you gave it and the document it is in. The next time you perform a 'connect link' operation, it creates a source anchor that refers to that target anchor.

See the chapter *Links* in the Reference Guide for more information.

Images

Web documents often include graphical images. Images are inserted in a document using an IMG element, which, like an A element, can be inserted through keyboard, menu or toolbar, or through dragging and dropping.

- Click on the View menu.
- If the menu contains the command Show Inline Images, choose this command. If it contains the command Hide Inline Images, do nothing.
- Move the insertion point just to the right of the ⟨/H1⟩ end-tag at the top of the document.

The following process is used to create an IMG within the H.i.P. Editor:

We will first walk through the process of creating an IMG within the H.i.P. Editor, and then use dragging and dropping to create an IMG.

- Click on the [image] toolbar button.

This inserts an IMG element, and brings up the Image Attributes dialog box.

- Click on the [Edit URL...] button.

This brings up the Edit IMG Source dialog box, which is similar (but not identical) to the Edit URL dialog box. In this dialog, you will construct a URL for the image.

- Click on the [Choose File...] button.
- Navigate to the folder *doc* in the H.i.P. Editor installation folder.

- Open the file *author.gif*.

The H.*i*.P. Editor has created the URL for this image from the file that you selected. Notice that the scheme and the path have been inserted.

Unfortunately, Hal didn't have a picture of Mike in digital form, as all the existing image files were deleted by the (then) art director in a fit of pique. So Hal had to substitute a different picture.

- Click on the [OK] in the **Edit IMG Source** dialog box.

- In the **Alternate Text** text box in the **Image Attributes** dialog, type:

    Anonymous

This provides text that a browser will display if it doesn't have the ability to display images, or if image-loading is turned off in a graphical Web browser. The H.*i*.P. Editor sets the HEIGHT and WIDTH attributes of the image automatically for you when you insert an image; this speeds image loading and page layout. In general, you do not need to change these numbers.

- Click on the [OK] button in the **Image Attributes** dialog box.

When you do this, a graphical image will be displayed inline, in the H.*i*.P. Editor document window. (This is not really Mike Woodhead.)

You can also drag and drop an image into the H.*i*.P. Editor. The image that you will create with drag and drop is identical to the image that you just created by inserting the IMG, so you may want to delete that image before creating this one.

- Switch to the Windows desktop or Windows Explorer and find the file *author.gif* in the *doc* folder under the H.*i*.P. Editor folder.

- Move and resize the document and application windows so that you can see the H.*i*.P. Editor window and and the *author.gif* file.

- Select the file *author.gif* by clicking and holding the left mouse button down on the file's icon.

- Continuing to hold down the left mouse button, move your mouse cursor to the H.*i*.P. Editor window. An image of the file you are dragging will appear under the mouse cursor. As the cursor is moved to the H.*i*.P. Editor window, it becomes a special drag and drop copy cursor. If you drag within a block of text, an insertion bar will

appear behind the cursor.

- Release the left mouse button wherever you want the IMG element to appear. An IMG element will be inserted, and the image will be displayed in the H.i.P. Editor document window if you have **Show Inline Images** turned on. If you release the left mouse key at a point where an image would be invalid, nothing will happen. You'll have to drag and drop the file again to a valid location.

> **Note** If you drag and drop to a saved document, the URL of the image will be in relative format. If you drag and drop to an unsaved file, or insert an image using 'Insert Element' or the image toolbar button, it will be a complete URL. See the Links chapter of the Reference Guide for more details on relative and complete URLs.

The image can be displayed in a graphical Web browser (if you have image loading turned on).

- Save the file.
- Click on one of the preview toolbar buttons, choose **Preview...** from the **File** menu, or type ⌈ Ctrl-M ⌋ at the keyboard.

For more information on images, see the chapter *Working with images* in the Reference Guide.

Backgrounds and fonts

Many Web browsers let you set the color of certain elements in a Web document. You can change the color of the background, the text, and the 'hot text', as well as the size of the fonts in your document. The H.i.P. Editor allows you to choose and change colors and sizes quickly and easily.

> **Note** Background images, background colors, and font colors and sizes, while commonly used, are not part of the HTML 2.0 specification, and are therefore not supported by all Web browsers. Use these features with caution, and make sure that your page looks acceptable without them.

You can create emphasis on Web pages by changing the font size and color. Go back to the second paragraph in the biography; the one that begins, 'PJ's untimely death...'.

Font size

You can change the font size by doing the following:

- Highlight the words 'offshore trust'.

- Click and hold on the [toolbar button] toolbar button, and select '+1' from the pop-up menu that appears.

The text that you highlighted is now surrounded by the FONT element, and the text in this selection will get bigger. All text starts with a default arbitrary size of '3'. This pop-up menu allows you to add or subtract from the base size of the font in the range 0-7. By choosing '+1' from the pop-up menu, we have changed the text size to '4'; that is, we have made the font size larger. These font size values do not refer to a specific font size, but are instead relative to the default font size.

If you want to set the font size for the whole document, you should use the BASEFONT element.

Text color

You can change the text color for a part of the document.

- In the same paragraph of the Biography, highlight the text 'throbbing headache'.

- Click on the [toolbar icon] toolbar icon.

The standard Windows color chooser will appear.
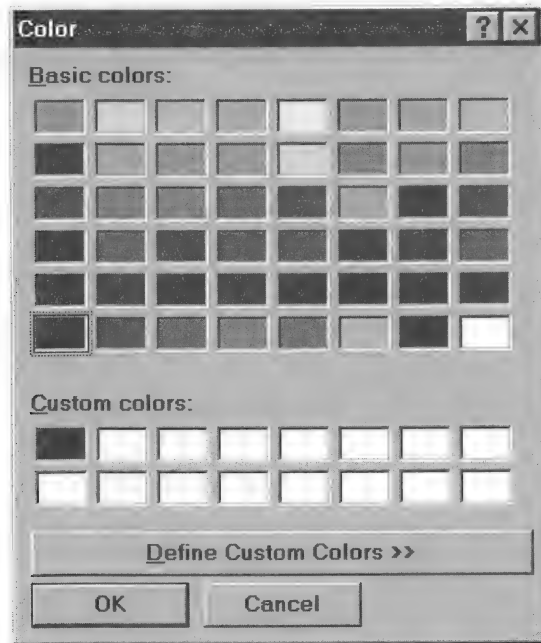
You can either select one of the 48 colors that are defined already, or make a custom color of your own by clicking on the [Define Custom Colors >>] button and manipulating the color square that appears. To make the text red:

- Click on the red square in the color chooser.
- Click on [OK] in the color chooser.

Notice that the text you highlighted is now surrounded by a FONT element, and the text color within that element has changed to reflect your color choice. To see it, click elsewhere to remove the selection highlight.

Background color

The background color of this Web page can be changed for emphasis and artistic effect. Let's set the background color to a light blue.

- Choose **Document Colors...** from the **Format** menu. This brings up the **Document Colors** dialog box which lets you set the background color.
- In the dialog box, click on the [Choose...] button next to the **Background** text box. This brings up the standard Windows color chooser.
- Click on the light blue square in the top row of the color chooser.
- Click on the [OK] button.
- Click on the [OK] button in the **Document Colors** dialog box.

The background color of the H.i.P. Editor document window will change to light blue.

Default text color

You can change the default color of the text and the links in your Web document using the **Document Colors** dialog box. Text and link colors are set in the same way as the background color: by bringing up the standard Windows color chooser. color chooser. Setting the text color changes the color of all the text in the document. Let's change the text color of the document to a dark blue.

- Choose **Document Colors...** from the **Format** menu. This brings up the **Document Colors** dialog box, which lets you set the text color.
- In the dialog box, click on the [Choose...] button next to the **Text** text box. This brings up the standard Windows color chooser. color chooser.
- Click on the dark blue square in the fourth row of the color chooser.
- Click on the [OK] button.
- Click on the [OK] button in the **Document Colors** dialog box.

The text color will change to blue. Notice that the text inside the FONT element that you inserted previously stays red.

Link colors

The **Document Colors** dialog box lets you set three different link colors. **Links** sets the color of a link that has not been visited. The default color on most Web browsers (and the H.i.P. Editor) is a bright blue. **Visited links** sets the color of a link that has already been visited. The default on most Web browsers is purple. **Active links** sets the color that links change to when they are clicked on. The default color on most Web browsers is red. You can set those colors in the same way that you just set the text colors.

Now you can see how this colorful document would look in a Web browser:

- Save the file.
- Click on one of the preview toolbar buttons, choose **Preview...** from the **File** menu, or type ⟨ Ctrl-M ⟩ at the keyboard.

If your Web browser supports both background and text colors, this document will be displayed with a light blue background, dark blue text, and one sentence in red.

Alignment

You can change the alignment of paragraphs and headers in several different ways, and the H.i.P. Editor gives you toolbar button access to the three most useful alignment features.

> **Note** Alignment of paragraphs and headers, while commonly used, is not part of the HTML 2.0 specification, and is therefore not supported by all Web browsers. Use this feature with caution, and make sure that your page looks acceptable without it.

In your document, create a new paragraph and type the following:

```
Another famous quote from
Mike Woodhead:  "I care, deeply,
about the welfare of my employees."
```

Put your insertion point inside the paragraph that you just created. Click on the ▦ toolbarbutton. This paragraph is now center-aligned. Clicking on the ▦ or ▦ button will change the alignment of the paragraph to 'left' or 'right', and the paragraph's position will change to reflect that.

Any HTML element that has an ALIGN attribute can be aligned in this way with the toolbar buttons. This includes all headers, paragraphs, and images (though images don't accept center alignment). To see how this document would look in a Web browser:

- Save the file.
- Click on one of the preview toolbar buttons, choose **Preview...** from the **File** menu, or type [Ctrl-M] at the keyboard.

If your Web browser supports these alignment features, the paragraph that you typed in will be centered.

Hal has now finished with the Mike Woodhead document. And he feels much better. Don't you? Time to close the file on Mike.

Find and Replace URLs

This section of the tutorial provides a short introduction to the **Find and replace URLs...** command. This command lets you convert a group of URLs that use a particular scheme, server, or location to URLs that use a different scheme, server or location. The most common use of this command is to convert URLs with local file references to URLs with Web server file references.

- Choose **Open...** in the **File** menu.
- In the dialog box that appears, navigate to the **Tutorial** folder inside the **Help** folder. (The default location is *C:/Program Files/SoftQuad/ HiP/Help/tutorial/*).
- Open the document called *tutor2.htm*.
- If the command **Show URLs** is present in the **View** menu, choose this command. If **Hide URLs** is present, do nothing. This ensures that the URLs are displayed in the document window.

This document contains six URLs, each pointing to a local file. Each URL starts with '*file:///c\/hal*'. In this exercise, you will change all of these URLs to refer to (fictitious) files on a Web server.

- Put the insertion point somewhere near the top of the file.
- Choose **Find and Replace URLs...** in the **Tools** menu.

The **Find and Replace URLs** dialog box appears.

```
H.i.P. Editor - Find and Replace URLs          [×]

Change URLs From: [                              ]

           To: [                              ]

  [ Find Next ]   [ Replace ]   [ Replace All ]   [ Cancel ]
```

- Pin this dialog box by clicking on the title bar with the right mouse button and then choosing **Pin** from the pop-up menu that appears.

- Enter the following in the **Change URLs From** text box (overwrite the default contents):

  `file:///c||hal`

- Enter the following in the **To** text box (overwrite the default contents):

  `http://www.server.com/~hmallard`

- Click on the [Find Next] button.

The H.*i*.P. Editor puts the insertion point inside the first element whose URL starts with '*file:///c\hal*'.

- Click on the [Replace] button.

The text '*file:///c\hal*' in the first URL changes to '*http://www.server.com/~hmallard*'.

You can use **Find and Replace URLs…** to modify several URLs at once.

- Click on the [Replace All] button.

The same replacement is performed in the remaining URLs.

Find and Replace URLs… is a form of **Find and Replace** command, but:

- It applies only to URLs.
- It can match text only starting at the beginning of the URL.

If you want to change something in the middle of a URL, you will have to enter everything that precedes it in the **Find and Replace URLs** dialog. For example:

- Enter the following in the **Change URLs From** text box:

  `http://www.server.com/~hmallard`

- Enter the following in the **To** text box (overwrite the default contents):

  `http://www.server.com/~aoutka`

- Click on the `Find Next` button.
- Click on the `Replace` button.

The H.*i*.P. Editor changes the text 'hmallard' to 'aoutka' in the current URL. For more information on the **Find and Replace URLs...** command, see the *Changing your URLs for your intranet* section in the *Links* chapter in the Reference Guide.

Forms

Hal also needed to learn about forms. There are certain HTML elements that can accept input from a user. A browser will display these elements as graphical objects, such as text boxes or drop-down lists. A 'form' in an HTML document is a set of such elements that lets the user enter some information and call a program, located on a Web server, that processes the information. For example, you could create a form that lets a user order a product that you're selling: you can set up the form so that when the user clicks on a 'submit' button, the order is sent to your order-processing program.

To implement this, you have to:

☐ Create the form(s).

☐ Install on your server the program that will process the form's data.

The second of these two steps is beyond the scope of the H.*i*.P. Editor. You will have to obtain supplementary documentation that explains this mechanism, which is known as CGI (Common Gateway Interface). If you open the Technical Reference page by choosing the **Technical Reference** from the **Help** menu, you will find several references to documents on this topic.

This tutorial explains how to properly set up a sample form. We suggest that for this exercise you create a new HTML document from a tutorial template.

- Choose **Open...** from the **File** menu.
- In the dialog box that appears, navigate to the **Tutorial** folder inside the **Help** folder.
- Choose the *tutor.htm* file.
- Now insert an **H1** element inside the **BODY** at the bottom of the document, and type:

  ```
  Buy my
  book!
  ```

- Insert a **P** element after the H1, and type:

  ```
  Just click on the "Submit" button in the
  form below to order any or all of these best-sellers
  at a fraction of the regular cost!
  ```

Form element

Now you're ready to start constructing a form.

- Click on the ▤ button to insert a **FORM** element after the **P** element.

This element encloses everything else in the form. For the remainder of this exercise, make sure that you're working inside the **FORM** element. The form element has attributes that specify its 'action'—how and where the form's data is to be submitted. We will specify the action at the end of the form tutorial.

The 'Form' toolbar—which contains the ▤ button—has many buttons. All of the form-related elements that you'll use in this tutorial can be entered using this toolbar.

Creating a text box

Now you'll enter the first element:

- Make sure the insertion point is inside the FORM element.
- Enter a P element and type:

    Name:

- Click on the [Abc] button on the 'Form' toolbar.

This inserts a text box into the form. the H.i.P. Editor's WYSIWYG form display gives you a good idea of what this form element will look like in your Web browser.

Now we must set the properties of this text field.

- Double-click on the text field, or click once on the text field to select it, and then choose Input Attributes... from the Forms menu.

This brings up a dialog box that lets you edit some of the properties of the current element:

Input Text Field	✕
Name:	
Value:	
Size:	Max. Length:
Apply	Reset	Cancel

This dialog box allows you to set the most important properties of this INPUT element: the name, value, size, and maximum length.

- In the Name text box type:

    cust-name

This value is used when the browser sends the form's data to the server, in order to identify which text box, drop-down list, etc., a particular piece of data came from.

Note the following other properties, which can enter values for if you like:

– The **Size** specifies the text box's size in characters. Entering a number here will change the size of the text box. If no number is entered, the text box will be 20 characters long.

– If you enter some text in the **Value** text box this text will be displayed in the text boxby default.

– The maximum length refers to the maximum number of characters that can be entered in the text field in a browser.

Entering several lines of text

A text box in a form lets you enter just one line of text. If you need to let your users enter several lines of text at once (to enter an address, for example), you should use the **TEXTAREA** element.

■ Insert a new **P** element and type:

    Address:

■ Click on the ▦ button to insert the **TEXTAREA** element. This is a graphical representation of a multi-line text area, which looks similar to the way this element is displayed in most browsers.
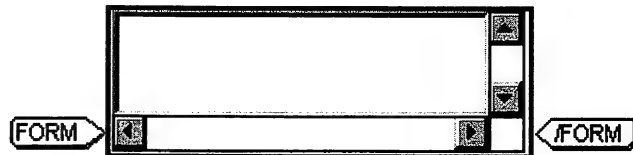
■ Double-click on the text area, or click once on the text area to select it, and then choose **Input Attributes...** from the **Forms** menu. This brings up a dialog box that contains the most important features of the **TEXTAREA** element: **Name**, **Default Content**, **Width**, **Rows** and **Text Wrapping**.

Multi-line Text

Name: cust_addr

Default Content:

Width (Chars.): 40   Rows: 5

Text Wrapping:

OK   Reset   Cancel
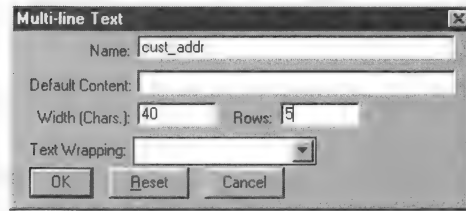
- Enter the following values in the **Multi-line Text** dialog box:

    Name: cust_addr

    Width: 40

    Rows: 5

    Rows and **Width** specify the dimensions of this input field: 5 lines deep and 40 characters wide.

- Click on the [OK] button.

If you want a multi-line text area to contain some default text, enter it in the **Default Content** text box.

Now you may want to see how the browser displays this element:

- Save the file.
- Click on one of the preview toolbar buttons, choose **Preview...** from the **File** menu, or type [Ctrl-M] at the keyboard.

The browser generates a multi-line field, which may also have scroll bars.

Presenting a selection list

Sometimes you will want the user to make one choice from a list of choices. In this example you'll see how to represent this with a drop-down list or scrollable list.

- Insert a new P element, and type:

    Credit Card:

- Click on the [▦] toolbarbutton to insert a **SELECT** element.

A graphical representation of the **SELECT** element will appear.

To edit the properties of the SELECT element:

- Double-click on the selection list, or click once on the list to select it, and then choose Input Attributes... from the Forms menu.

The Selection list dialog box appears, allowing you to create and edit a list of choices.
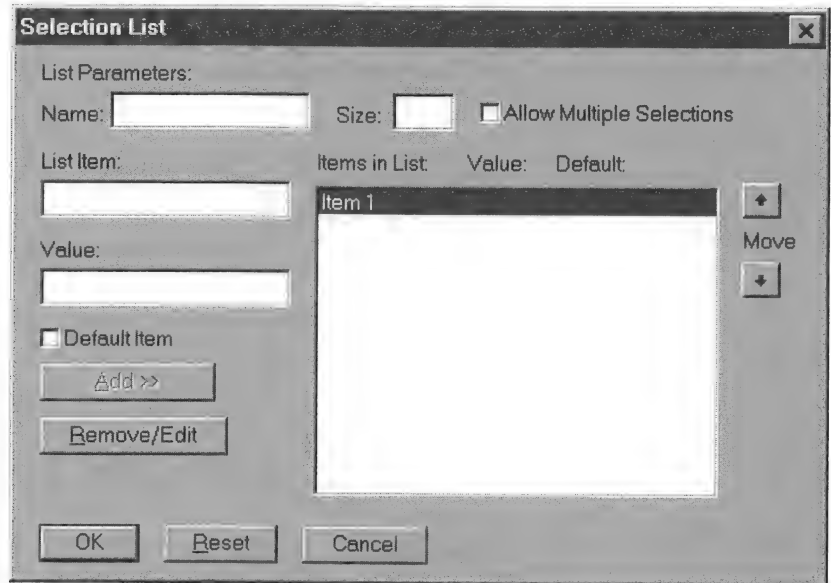
This dialog box has two purposes: it lets you set the properties of the selection list and create and edit items within it. First, let's edit the properties of the list:

- In the Name text box, enter

  card-name

  This is the identifier for the SELECT element.
- In the text box labeled Size, enter '1'. This means that only one option will be shown at a time in the selection list.

The H.i.P. Editor puts one item in the selection list by default. You can edit the default list item and create new ones. To edit the default list option:

- Highlight the words 'Item 1' in the list of items, and click on the [Remove/Edit] button. The list option will 'jump out' of the list and become editable.

- Highlight the words 'Item 1' in the List Item text box and enter:

    Visa

- Click on the [Add >>] button. The item you just edited will move back into the list.

This list item (represented in HTML as an OPTION element) represents one choice in the selection list. The text that you typed will appear as the name of an option in the selection list.

- To add a new list option, move the insertion point to the List Item text box and type:

    MasterCard

- Click on the [Add >>] button.

- Move the insertion point to the List Item text box and type:

    Amex

- Click on the [Add >>] button.

- Click on [OK] .

To see what this looks like in the browser:

- Save the file.

- Click on one of the preview toolbar buttons, choose Preview... from the File menu, or type [Ctrl-M] at the keyboard.

- In the browser, click on the selection list.

Notice that each list item that you entered as a list option in the Selection List dialog box is a choice in the list.

Selection list properties

□ You can add a **Value** for each list option, which would then be sent with the form data instead of the **List Item** text of the list option. This is not necessary, though; if there is nothing entered in the **Value** field, the name of the list item will be sent when selected.

□ If you wanted, you could have represented this selection list as a scrollable list rather than a drop-down list. To do this, you would set the SIZE attribute of SELECT to 2 or greater by typing the value into the Size text box; this value specifies how many list items are shown at a time.

□ If you want to be able to choose more than one item from this kind of list, turn on the **Allow Multiple Selections** check box.

□ To set one list option as the default (which is displayed highlighted in most Web browsers), edit that list option and turn on the **Default Item** check box.

□ To change the order of list options in a drop-down list, select a list option, and use the [↑] and [↓] buttons to change that item's position in the list.

Just to complete this section of the form:

■ Insert a new P element after the last one, and type:

  `Card number`

■ Insert a text box by clicking on the [Abc] toolbar button.

■ Double-click on the text field, or click once on the text field to select it, and then choose **Input Attributes...** from the **Forms** menu.

■ Set the NAME attribute to 'card-num'.

■ Click on the [OK] button.

Check boxes

There are several other different kinds of input elements that can appear in forms. One of these is *check boxes*: you would create this kind of box if you wanted the user to make a 'yes/no' choice. These are different from *radio buttons* (the next page), which you would use if you want the user to make one choice from a group of choices.

- Place your insertion point inside the ⟨/FORM⟩ end-tag.
- Insert a new P element after the last one, and type:

      Check one or more titles:

- Insert a new P element after the last one, and type:

      The Dentistry of Frederic Chopin

- Insert a check box by clicking on the 🗹 toolbar button.

The H.i.P. Editor inserts a check box icon.

- Double-click on the check box, or click once on the check box to select it, and then choose **Input Attributes...** from the **Forms** menu.
- In the dialog box that appears, type chopin in the **Name** text box. You don't need to enter anything in the **Value** text box.
- Click on the ⎡ OK ⎤ button.

Now add two more titles in the same way:

- Insert a new P element after the last one, and type:

      Motors and Such by Eddy Schneider

- Insert a check box.
- Double-click on the check box, or click once on the check box and then choose **Input Attributes...**.
- Type motors in the **Name** text box.
- Click on the ⎡ OK ⎤ button.
- Insert a new P element after the last one, and type:

      HTML for Travellers

- Insert a check box.
- Double-click on the check box, or click once on the check box and then choose **Input Attributes...**.
- Type html in the **Name** text box.

You can turn a check box 'on' by default, so that it will appear checked when viewed in a browser or in the H.i.P. Editor:

■ Click on the **Checked?** box.

■ Click on the ⌈ OK ⌉ button. Notice that the check box changes in the H.i.P. Editor to reflect its checked status. You can have more than one box turned on by default.

To see what this looks like in the browser:

■ Save the file.

■ Click on one of the preview toolbar buttons, choose **Preview...** from the **File** menu, or type ⌈ Ctrl-M ⌉ at the keyboard.

Try clicking on the buttons with the mouse. You can turn on all, any, or none of the buttons.

When the form is submitted, the default value associated with turned-on check buttons is the word 'on' (turned-off check boxes are not submitted to the server). If you want to override this default, enter the text to be associated with the check box in the **Value** text box of the check box dialog.

Radio buttons

A group of radio buttons lets the user make one (and only one) choice from a group of choices.

■ Make sure your insertion point is within the ⟨/FORM⟩ end-tag.

■ Insert a new paragraph after the last one and type:

    `Preferred language:`

■ Insert a new paragraph after the last one.

■ Insert a radio button by clicking on the ▣ toolbar button.

■ Double-click on the graphical radio button element, or click once on the radio button and then choose **Input Attributes...** from the **Forms** menu.

■ In the dialog box that appears, enter the following text into the text boxes indicated:

    – Enter 'language' in the **Name** text box.

- Enter 'english' in the **Value** text box.
- Click on the **Checked?** box to set this radio button as the active one. Notice that the radio button changes in the H.i.P. Editor to reflect its active status. Only one radio button in a group can have the **Checked?** box activated.

- Click on the ⌐OK⌐ button.
- Move the insertion point to the right of the radio button and type:

  `English`

Now add two more choices in the same way (keep all three choices in the same paragraph):

- Insert a radio button.
- Double-click on the radio button, or click once on the radio button and then choose **Input Attributes...** from the **Forms** menu.
- In the dialog box that appears:
  - Enter 'language' in the **Name** text box.
  - Enter 'french' in the **Value** text box.

- Click on the ⌐OK⌐ button.
- Move the insertion point to the right of the radio button and type:

  `French`

- Insert a radio button.
- Double-click on the radio button, or click once on the radio button and then choose **Input Attributes....**
- In the dialog box that appears:
  - Enter 'language' in the **Name** text box.
  - Enter 'spanish' in the **Value** text box.

- Click on the ⌐OK⌐ button.
- Move the insertion point to the right of the radio button and type:

  `Spanish`

Notice the following about the attribute values you've just entered:

– All three radio buttons have the same **Name**: 'language'. This puts them in the same *group*, which means that the browser will allow only one of these three to be checked at once. If the form contains a separate group of radio buttons, the **Name** for all its members will have to be something other than 'language'.

– The content of the **Value** text box is sent to the server if the corresponding button is turned on when you submit the form, thus telling the server which button in this group was turned on.

To see what this looks like in the browser:

■ Save the file.

■ Choose one of the preview toolbar buttons, select **Preview...** from the **File** menu, or type (Ctrl-M) at the keyboard.

Try clicking on the buttons with the mouse. You can turn on only one of the radio buttons at a time. The button labeled 'English' is initially turned on by default.

Reset

You can create a button that restores all the form's controls (text boxes, radio buttons, etc.) to their default values:

■ Insert a new **P** element after the last one (inside the ⟨/FORM⟩ end-tag).

■ Insert a reset button by choosing the [⊞] toolbar button.

■ Double-click on the reset button, or click once on the reset button and then choose **Input Attributes...** from the **Forms** menu.

■ In the dialog box that appears, enter 'Reset to defaults' in the **Value** field.

■ Click on the (OK) button.

Notice that this changes the text on the reset button in the H.*i*.P. Editor. To see how the reset button works in a Web browser:

- Save the file.
- Choose one of the preview toolbar buttons, select **Preview...** from the **File** menu, or type (Ctrl-M) at the keyboard.
- Enter some data in the form.
- Click on the [Reset to defaults] button in the form.

The form's controls revert to their default values. The text boxes are all blank and the check boxes are turned off. The radio button labeled 'English' is turned on.

Submit

Your form is almost complete. All you need to do is create a button that causes the browser to submit the form.

- Next to the reset button, insert a submit button by choosing the ░░toolbar button.
- Double-click on the submit button, or click once on the submit button and then choose **Input Attributes...** from the **Forms** menu.
- In the dialog box that appears, enter 'Submit order' in the **Value** text box.
- Click on the [OK] button.

Notice that this changes the text on the submit button in the H.*i*.P. Editor. To see how this looks in a Web browser:

- Save the file.
- Choose one of the preview toolbar buttons, select **Preview...** from the **File** menu, or type (Ctrl-M) at the keyboard.

Setting the action of the form

You have now finished setting up the form's elements. All that remains is to set the 'action' of the form; that is, what happens to the data entered by the user when the submit button is pressed.

- Place the insertion point immediately to the right of the opening FORM⟩ tag. Choose **Edit URL...** from the **Markup** menu.

The URL you'll create is referred to as the form's 'action'.

- From the **Scheme** drop-down list, choose 'http'.
- In the **Host** text box, type:

  www.sq.com

- In the **Path** text box, type:

  cgi-bin/quagmire

- Click on the ⟨ OK ⟩ button.

The 'action' you've just specified refers to a program, located on SoftQuad H.i.P.'s Web server, that can process the data entered in the form. In the next section of the tutorial, you can submit the form to this program. Normally you would specify a program on your own server, though in fact you can specify programs located anywhere on the Web.
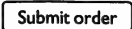
- Choose the **Element Attributes...** command from the **Markup** menu, right-click and choose **Element Attributes...** from the pop-up menu that appears, or type ⟨ F6 ⟩ at the keyboard.

This brings up a dialog box that lets you edit the *attributes* of the current element.

- Set the attribute called METHOD to the value 'GET' (it may have that value already).
- Click on the ⟨ Apply ⟩ button.

Submitting the form

Preview the document in your Web browser.

- Enter some data in the form. (Don't enter a real credit card number.)
- Click on the [Submit order] button in the form.

If you have an active connection to the Internet when you are previewing your H.i.P. Editor document, the form will now be submitted to a program on SoftQuad's Web server. This program doesn't actually process an order, it just echoes back the information that the browser sent it. This information will appear in the browser window: you can return to the form by clicking on the button that takes you to the previous document.

The information is presented in pairs containing a 'name' (corresponding to the NAME attribute of the text box, check box, etc.) and a 'value' (for text boxes or 'text areas', this will be the data you typed in; for check boxes the value 'on' is submitted—check boxes that aren't turned on are ignored; for radio buttons the value of the VALUE attribute is sent). You may notice a couple of unusual things about the format of the text: spaces are replaced by a '+' sign, and some special characters (notably newline, '=', and '&') are replaced by '%nn', where the n's are digits from 0-9 and/or letters between 'A' and 'F'. This is the standard format that browsers use for submitting data to the server.

Mailing the contents of a form

If you want to use forms, but don't have a program set up on your server to process the data, you can send the raw content of the form to an e-mail address, which is a quick and easy way to get form functionality. However, the content of that e-mail may require editing. The form's action must point to an e-mail address, so that the content of the form will be e-mailed to that address when the submit button is pressed. To create a form that is mailed to a particular address:

- Place your cursor immediately to the right of the opening (FORM⟩ tag.
- Choose Edit URL... from the Markup menu.
- From the Scheme drop-down list, choose 'mailto'.
- In the Path text box, type your e-mail address.
- Choose the Element Attributes... command from the Markup menu, right-click and choose Element Attributes... from the pop-up menu that appears, or type (F6) at the keyboard.

- Set the attribute called METHOD to the value 'POST' (it may have that value already).
- Click on the [Apply] button.

If you have an active Internet connection at the time when you view this document in a Web browser, filling out this form and clicking the submit button will mail the contents of the form to the e-mail address specified.

For more information

There is a summary of what you have learned about forms, and some more information about advanced form editing, in the *Core HTML* chapter of the Reference Guide.

Index